DOCUMENT RESUME

ED 232 690 IR 050 338

AUTHOR	Ya-chun, Lian
TITLE	RAMAS: The RITL Automated Management System. Master Control and Periodicals Control Subsystems. Stockholm Papers in Library and Information Science.
INSTITUTION	Royal Inst. of Tech., Stockholm (Sweden). Library.
REPORT NO	TRITA-LIB-4076
PUB DATE	Nov 82
NOTE	88p.
PUB TYPE	Audiovisual Materials (100) -- Reports - Descriptive (141)
EDRS PRICE	MF01/PC04 Plus Postage.
DESCRIPTORS	Flow Charts; Foreign Countries; Higher Education; Library Administration; *Library Automation; Library Networks; *Library Technical Processes; Minicomputers; *Online Systems; *Periodicals; *Recordkeeping; Records (Forms); Research Libraries
IDENTIFIERS	*Royal Institute of Technology (Sweden)

ABSTRACT

An automated minicomputer-based library management system is being developed at the Swedish Royal Institute of Technology Library (RITL). RAMAS (the RITL Automated Management System) currently deals with periodical check-in, claiming, index-handling, and binding control. A RAMAS bibliographic record can be accessed from eight different points (e.g., International Standard Serial Number/ISSN, title key, full title), and the check-in display shows full information on recent issues (e.g., arrival dates, claiming dates, double issues). Claiming letters and binding notices are controlled by parameters characteristic of each journal. The RAMAS bibliographic file includes the full Swedish Union Catalog of Scientific and Technical Serials/Periodicals (LIST TECH) and can be continuously updated as an online reference tool. The bibliographic main file has been extended to accept transformed records from the Swedish National Research Library System (LIBRIS) and further development is in progress towards an integrated system which will be used for online acquisition, check-in, circulation, and catalog operations dealing with monographs, periodicals, serials, conference proceedings, and multivolume works. The report provides (1) an introductory preface by Stephan Schwarz; (2) an article describing RAMAS; (3) appendices outlining RAMAS data structure and record formats; (4) a brief survey of the RAMAS programs; and (5) flow diagrams for the RAMAS Master Control and Periodicals Control Subsystems. (Author/ESR)

****
* Reproductions supplied by EDRS are the best that can be made *
* from the original document. *
****

U.S. DEPARTMENT OF EDUCATION
NATIONAL INSTITUTE OF EDUCATION
EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)
This document has been reproduced as received from the person or organization originating it.
Minor changes have been made to improve reproduction quality.
Points of view or opinions stated in this document do not necessarily represent official NIE position or policy.

ED232690

REPORT TRITA-LIB-4076 NOVEMBER 1982

Stockholm Papers in Library and Information Science

RAMAS: THE RITL AUTOMATED MANAGEMENT SYSTEM

Master control and Periodicals control subsystems

Lian Ya-chun
Royal Institute of Technology Library
(on leave of absence from ISTIC, Beijing)

With a preface by Stephan Schwarz

"PERMISSION TO REPRODUCE THIS MATERIAL HAS BEEN GRANTED BY
S. Schwarz
TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)."

IR050338

EDS2500

Stockholm papers in library and information science
TRITA-LIB-4076, Royal Institute of Technology Library
Stockholm 1982, ISSN 0346-9042

PREFACE

by Stephan Schwarz

Computers have been successfully used in research libraries for over two decades, sufficient time for several generations of computers, general file handling systems and library applications software. A major effort was first spent on bibliographic databases associated with abstract journals and on catalogues, local or in library networks. It seemed natural to extend such efforts towards integrated library management systems, including online catalogues, computer aided acquisition, periodicals control and circulation management. These efforts have given rise to a large literature in library journals, even to specialized titles like: "Journal of Library Automation" and "Program".

Relatively early there was a recurrent dream of a "turnkey" package that could be bought ready-made or made to order, installed and run with little or no local expertise for maintenance. What was often not seriously taken into account was the combined effect of the rapid development of computers (and the ensuing short lifetime of software systems), the complexity of library management problems, and the relatively small and rather disparate demand. Development projects tended to drag along for a period covering several generations of computers. In spite of progress in standardization (e.g. bibliographic records, international book and serial numbers, etc), the requirements in different libraries have not been standardized, and with rising costs for skilled manpower in systems development the market is simply too small to attract much interest on the part of software enterprises. Notable exceptions seem to be associated with the marketing of hardware (minicomputers) for which library systems are specially designed. On the contrary, systems for large computers tend to be special developments in academic libraries, or library groups, original or based on some more general file handling software. Such systems seem to be in permanent development, and I have not seen any of them marketed, probably due to the inconvenience of meeting contractual obligations within a non-commercially oriented organization.

To quote an authority (Dan Tonkery, UCLA in ref 1 p 6): "Serials control and especially serials check-in proved to be a far more complex programming problem than designers had originally projected". On the whole, it seems likely that the dream of a turnkey system, ready-made or made to order, that can be plugged in and run to satisfaction is not to come true without considerable restrictions in the near future. A certain warning against oversell as to functions, flexibility and portability, common in the library systems development area (ref 3), is warranted.

At RITL we have been interested in library management systems for minicomputers for several years, a program supported by DFI, the Swedish Delegation for S&T Information Provision. The latest developments indicate the possibility of an integrated system, comprising the management of an online catalogue, acquisition and periodicals control routines, circulation management and statistical/financial support operations. The design of the RAMAS system, presented in this report, is a step towards this objective. The work has been performed by a group with library and systems development competence, but has been made possible by the unique capacity of our research associate, Mr. Lian Ya-chun, from ISTIC (Institute for S&T Information of China) in Beijing. At present the Master Control System and the Periodicals Control System are operational. The acquisition routines have been analysed and a system interacting with the Swedish national research library catalogue (LIBRIS) will be implemented. It is now investigated whether the circulation system, developed for the ALIS system at the National Technological Library of Denmark can be adapted to our computer configuration.

REFERENCES

1) W.G. Potter and A.F. Sirkin (eds): Serials Automation for Acquisition and Inventory Control (American Library Association, Chicago 1981)

2) P. Gellatly (ed): The Management of Serials Automation - Current Technology and Strategies for Future Planning (A Monographic Supplement to The Serials Librarian 6 (1981/82), The Haworth Press, New York, 1982)

3) S. Schwarz: The Small Systems Syndrome and the Inverted Binocular Strategy - some questions in search of answers (t.b.p. in Journal of Documentation, March 1983)

4) Lian Ya-chun: Introduction to RAMAS - the RITL Automated Management System (t.b.p. in the Proceedings of the International Conference on the Application of Mini- and Micro-computers in Information, Documentation and Libraries, Tel-Aviv, Israel, March 13-18, 1983).

TABLE OF CONTENTS

I	Preface by Stephan Schwarz		3
II	RAMAS, The RITL Automated Management System by Lian Ya-chun		7
III	Appendices		
	1.	Bibliographic Master File Data Structure	38
	2.1	Bibliographic Image of Periodical	39
	2.2	Bibliographic Image of Monograph	39
	3.1	Access Image	40
	3.2	Check-in Image for Periodicals	40
IV	A brief survey of the RAMAS programs		41
V	Flow Diagrams (table of contents)		49
	1.	RAMAS Master Control Subsystem	51
	2.	RAMAS Periodicals Control Subsystem	66

RAMAS
THE RITL AUTOMATED MANAGEMENT SYSTEM

by Lian Ya-chun
Royal Institute of Technology Library
(on leave of absence from ISTIC, Beijing)

ABSTRACT

Periodicals control is a major problem for all large research libraries and in spite of massive efforts in developing computer-assisted library management routines, this complex and tedious operation is mostly performed by unsatisfactory traditional techniques. An automated minicomputer-based system has been developed at the Royal Institute of Technology Library, which caters for most of the problems related to check-in, claiming, index-handling and binding control for periodicals. The bibliographic record can be accessed from 8 different points (ISSN, Title key, Full title etc) and the check-in display shows full information on recent issues (arrival dates, claiming dates, double issues, etc). Claiming letters and binding notices are controlled by parameters characteristic of each journal. The bibliographic file includes the full Swedish Union Catalog of S&T Serials/Periodicals (LIST TECH) and can be continuously updated as an on-line reference tool. The bibliographic main file has been extended to accept transformed records from the Swedish National Research Library System (LIBRIS), and further development towards an integrated system (Catalogue, Acquisition, Serials/-Periodicals Control, and Circulation/Loans) is in progress.

摘　要

对所有大型图书馆系统，期刊管理是一个主要的问题。以往在研制计算机辅助的图书馆管理系统时，尽管时此作了很大努力，但这一复杂而又乏味的工作仍靠以传统的手工来进行。瑞典皇家工学院图书馆已研制出采用小型计算机的自动化系统，它配备了期刊管理中涉及到的登录、索赔、索引处理及装订方面种功能，能从8个角度（包括ISSN、题目关键词、题目全部内容）查找数据库中的文献记录。终端可以显示出最近若干期（最多16期）的全信息（如收到日期、索赔日期、合刊情况）。根据每种杂志的情况，系统能自动打印出索赔信件及装订通知。数据库的内容包括全瑞典种数期刊、杂志联合目录（即LISTTECH），并能通过联机方式对此加以更新。此外，还包括从瑞典国家图书馆系统（LIBRIS）移植过来的记录。本书发展成为一套完整的、包括分类、采购、期刊管理及借阅的系统。

INTRODUCTION: GENERAL OBJECTIVES AND CONDITIONS

RAMAS (RITL Automated Management System) is a mini-computer based tool for research libraries developed at the Royal Institute of Technology Library in Stockholm. It is designed to contain subsystems called "Master control", "Serials control", "Circulation" and "Acquisitions", which can be used for online acquistion, check-in, circulation (loans) and catalog operations dealing with monographs, periodicals, serials, conference proceedings and multivolume works. In addition, modules for special applications are foreseen: statistic analysis, binding, claims, accounting, etc. At the moment, the Master control and Serials control subsystems have been developed and implemented, the latter including claiming control and binding control.

The present RITL minicomputer configuration is based on a DEC PDP 11/34. The following features had to be considered in designing and building RAMAS:

1) the existing equipment should be used
2) the system should be user-oriented, i.e. easy to operate
3) the system should be "transparent", i.e. easy to maintain and improve
4) operations should be based on online access to databases
5) the system should allow access by several users simultaneously

To satisfy these requirements, all applications programs are written in BASIC-PLUS-2 with RMS and FMS, running under RSX-11M on a PDP 11/34 computer with 256 K (16 bit) word memory. Software is maintained on one RM02 (67 MB) disc. The systems are accessed through VT-100 terminals. RMS (Record Management Services) is a set of software routines that processes data in units called records between a running program and an operating system. FMS (Form Management System) is the tool for developing form applications. Using FMS, which allows accessing and entering data through a predesigned format on the screen, not only more convenience is achieved for the user-machine interaction, but it also allows for more flexibility in the systems design.

1. MASTER CONTROL SYSTEM

From the point of view of systems design it seems preferable to construct a universal record format and a file to which various kinds of records can be added. This approach requires less application programs and simpler management software than the alterative of maintaining a variety of different files. We will call a record defined in accordance with our format requirements a "main bibliographic record" or a master record. On the basis of this record format, all bibliographic and supplementary information related to books, journals, serials, proceedings etc in the library can be recorded in a bibliographic main file called "BIBLMAIN.FIL". This is an indexed file of variable length structured records. Each record contains up to a maximum of 36 data fields. By means of the Form Management technology, the entire information in one record can be displayed on a special image on the terminal screen. This display is also used for entering new records online, and for entering or changing information in any particular field.

1.1 Master record structure

The master record format is given on Appendix 1. The first 8 fields are indexing keys through which one can quickly access the record. The first key, the record number, is the "primary" key in the system. The seven other alternative keys comprise Title key, Author key, Conference name key, ISSN/ISBN/NBN, Signum, Classification and Type of literature. For indexing purposes, it is necessary to have a unique serial number in each record. In RMS-11 all the keys except the primary key can be changed. Since the serial number does not have to be changed, we choose to treat it as the primary key. The eight indexing keys together represent the fixed-length fields of each record. The total fixed length equals 54 bytes.

A special technique is used to save space when representing the first key (record number) and the seventh key (classification). RMS-11 permits indexing on a string constructed from any of 128 alphabetic, numeric and control characters which can be represented in ASCII codes. In order to save

storage space on the disc, the record number is represented by a 3-byte ASCII code based on 128. This makes 128 · 128 · 128 numbers available, i.e. more than two million (exactly 2097151).[x] In this representation the computer, for example, sees the record number 545450 as " ! % * " and the number 200100 as " z8T ". However, the external representation of the record number is in decimal form.

The external form of a classification code has a maximum length of 19 characters which looks, for example, like 060.456 326 109 453. The full stop and spaces occupy regular positions since the digits appear in groups of three. Therefore, the computer can do without them and only take the effective numeric numbers, converting the string into a 7-character ASCII string as internal data format. Using a standard conversion routine, it can reconstruct to original representation when it needs to be displayed.

Other items making up a record are variable-length text fields (see Appendix 1). These fields are divided into 10 sections, some of which join more than one field, while some consist of only one. In the case of more than one field in a section, only the last field has a variable length, all the others should be of fixed length. Every section has been given a pointer number, and the number within brackets shown in App. 1 stands for the maximum length of that section. Some data fields associate with different contents for different types of literature. There are, for example, "Libraries" in the periodicals case, and the same field is used for the "Originator" in the case of monographs or collective volumes, but both of them have the same section pointer number. The maximum length of the title field or conference name field is 256 characters. If the actual length of a title or conference name exceeds 256 characters, the exceeding parts can be put into the comments field, wich also has a maximum size of 256 characters. The 256 characters are considered to be enougth for most cases. The present limit of 256 is set by a 2-digit counter number in the pointer field, expressed in a sexadecimal-digit system (16 16=256). Changing to a 20-digit representation system, for example, one can get a limit of 400 characters.

---

x) The modulo of the ASCII values in PDP 11 BASIC+2 is 256, but we use modulo 128. In fact, the represented maximum for a 3-byte ASCII code should be 16777216.

The last field named "Mark" in the record is used for flagging data in case more than one signum or classification have to be entered (or have been entered) to complete a bibliographic record.

Although the maximum length of the record equals 1361 (54+64+256+256+ +71+71+62+256+146+21+103+1) characters, a case in which every field is used to its maximum length is actually very rare. In most cases, the effective record length will be less than 1000 characters. Restrictions presented by the computer capacities only allow for a maximum record length of 974 characters at present. With improved computer capacity, it may be expected to increase to considerably more than 1000 characters.

On the terminal image, only two lines of fixed length have been allocated for display of Monograph Title or Conference Name, or for Comments (see Appendix 2). A text of more than 160 characters can not be completely displayed on the related area at one time. But by means of an FMS special technology - a scroll method - using the "↑" or "↓" function key on the keyboard one can make the string intervals (1st--80th, 81st--160th or 161st--240th, 241st--256th character) appear in sequence.

1.2 Search keys

The RMS-11 software only allows access to data through primary key or alternative keys. Although the maximum length of a key (string) can be up to 256 characters, it is not practical to use the whole title or full name as a key instead of a reduced "Title key" or "Author key", since it would be a waste of disc space without particular advantage.

In the present version, the Title key consists of maximum 6 characters. The following rules of forming a key are applied (as suggested in the Zürich (ETH) computerized system Pekos-III):

1) For a one-word-title, take the first 6 characters as the key, e.g.
Electronics.....ELECTR.

2) For a two-word-title, take the first 3 characters from each word as the key, e.g.
Physics today.....PHYTOD.

3) For a three-word-title, take the first 2 characters from each word as the key, e.g.
Journal of physics.....JOOFPH.

4) For a title with more than three words, take the first character from each word up to six characters, e.g.
Library services in transition.....LSIT
Bulletin of the atomic scientists.....BOTAS
Advances in nuclear science and technology.....AINSAT
International journal of mass spectrometry and ion physics.....
IJOMSA

5) Use only the abbreviated title if it appears at the beginning of a title, e.g.
ISA transactions.....ISA
JETP letters.....JETP

6) Note that the general rules apply also in case the result is a code with less than 6 characters, e.g.
Physics letters B.....PHLEB

7) The key can use both upper case and lower case or mixed strings, e.g.
Physics letters B.....PHLEB or phleb or PhleB

8) The rule of forming a Conference Name key follows the same principles as the rule for Title.

9) The Author key also consists of maximum 6 characters. It contains 5 characters from the last name (family name) plus the first character (initial) of the first given name. This rule applies also in case of short family name, where full family name and initial are used accordingly. If access starts at full author name, either of the following two rules applies (and the same code will be generated):

a) last name is given first, then a comma and finally the given name (only one), e.g.
Einstein, A (or Albert).....EINSTA

b) given name(s) first then family name; spaces between names and not more than three names, e.g.
Albert Einstein.....EINSTA
Karl Raimund Popper.....POPPEK

1.3 Master Bibliographic File

The present periodicals part in the Master Bibliographic File has been created by transfer directly from the "LIST TECH" file (the RITL union catalog of current S&T periodicals in Swedish research libararies). At present, more than ten thousand records have been transferred. The LIST TECH file is a terminal-format file that stores ASCII characters. The record contains a maximum of 18 data fields. Among these, we only pick up 11 items, marked with an asterisk (*) (see Table 1) for transfer from LIST Tech to the present Master File. Other supplementary data can be added online from the terminal. Transfer of bibliographic records from the LIBRIS file is in the process of development. Of course, the Master Control Subsystem offers the ability to create new records directly. Two other separate index files are linked to the Master Control System, viz. the "Multiple Signum and Classification" file and the "Subtitle Key" file.

Table 1:
LIST TECH Data Fields

Data field code		Data Name
001		Record number
020	*	Signum (RITL identification code)
030	*	Private libraries codes
040	*	Public libraries codes
044		Libraries included in LIST METAL
080	*	Publisher location (city)
110	*	ISSN
200	*	Title
330		CODEN
810	*	Main Record type mark
820		Source for acquisition
830	*	Periodicity
840		Index mode information (registration)
850	*	Separate cataloguing mark
940		General hints
950		Price
970	*	Location within RITL
980	*	Holdings information (and information on bound volumes)

1.4 Functions and applications

The Master Control Subsystem allows online retrieval, uppdating and creating of new records.

1.4.1 Retrieval

The user opens access to the Master System by typing a message "RUN RAMAS". The system then displays an accession menu as a response (Appendix 3.1) and the cursor stays in the position of "Access point". This means that you can access the Master Bibliographic Data Record from 8 different points in retrieval, and an extra point for creation of new records:

1) Record number
2) Title key
3) Author key
4) Conference key
5) ISSN/ISBN/NBN
6) Signum
7) Classification
8) Entry of new record (not used in retrieval)
9) Full name

The selection command is the corresponding number and the RETURN key. If the number given is acceptable, the cursor will move to the corresponding position, otherwise it will stay at "Access point" and wait for a new command. For example, after typing 1 and RETURN, the cursor will move to the right side of "Record number". (On the other hand, on typing 10 and RETURN, nothing happens to the cursor.)

The next step is to type a relevant content for the access point selected (e.g. Title key or Signum etc) and ending by RETURN key. The system then begins to search. If there are no appropriate records retrieved matching the key given, the system will show a message "No items retrieved for this key" in the last line of the screen. After a while, a new access image will appear on the screen, and one can enter a new search request from any point.

If the search is successful, the system will first clean the whole screen, then display a bibliographic form (Appendix 2) without any data in it, and then pass the data into the appropriate data fields.

For some fields (e.g. Title), if the contents are more than two lines (160 characters), a message "Text is more than two lines" will be shown in the last line of the screen. Using "↓" key or "↑" key, one can see the whole contents within the two-line area on the screen (scroll technique).

Except for the case of accessing by record number, more than one record may match the same search key. If that happens, a numbered list giving the country of origin and title will be displayed. Typing the number of the record one wants to look at, the system will immediately display the whole information concerning this selected record in the bibliographic image. If one wants to select any other record on the list one can input "R" as a response the instruction G/R/U/E given by system, and then type the number (see below for meaning of symbols).

The RAMAS system permits accessing from full name (full title, full conference name or full author name). In this case, one has to type "9" in Access Point status and to give one of "a", "b" and "c" (standing for respectively Title, Author name and Conference name). The cursor will stop in the lower left corner of the screen. One can then input the full Title etc. If the content is longer than one line, the line terminator is Downarrow key "↓" instead of RETURN key. According to the rule of forming a key, it is enough only to input the first six words of the title or conference, since the rest is not used for identification of records.

Summing up, the retrieval process has the following flow (cf Figure 1):

1) Type "RUN RAMAS" to access system
2) From access format display: enter search key (1 to 7) or full name (9)
3) Select retrieved record number (if more than one record matches)
4) Display Master record information
5) Select and process instruction G/R/U/E, where
   "G" means go back (i.e. to 2)
   "R" means repeat (i.e. 3,4,5)
   "U" means update
   "E" means program exit

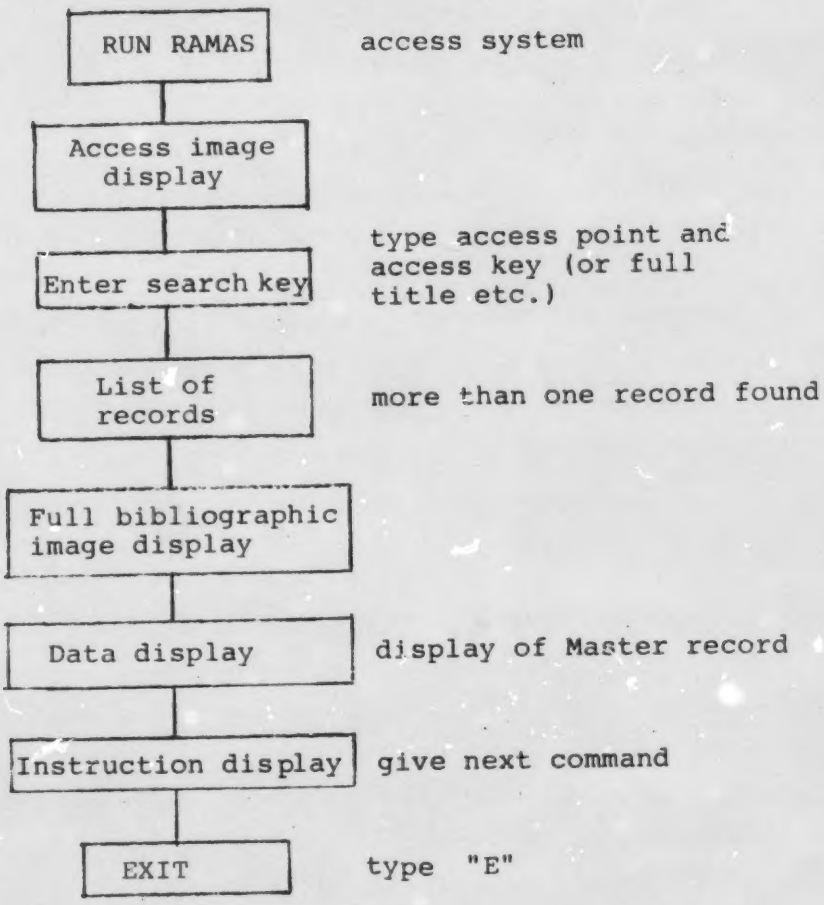


Fig 1: Retrieval flow diagram in Master control

1.4.2 Updating

Since updating a record depends on previous successful search, first one must find and display the record which needs updating. This can be done by the same procedure 1) to 5) as in retrieval. The difference is only that, when the selection instruction appears at the last line of the screen, one must press the "U" (update) key in procedure 5) and, with the "TAB" or "RETURN" key, move the cursor to any data field (except Record Number) that needs to be changed. Within a data field, after pressing the "PF1" key on the keypad,

one can perform insertion or deletion to the text. To delete an entire field one can press the "LINE FEED" key.

When the cursor is in one of the fields Title, Conference or Comments one can get to the line desired only by means of the keys "↓" or "↑" and the "RETURN" key can only be used for ending operation of these data fields.

The keys for Title, Conference Name, Author, Variant Title, and Serial Title are changed internally by the program when related full text strings have been changed.

To re-check the data when the cursor moves out of the last field on the screen, one can press the "BACK SPACE" key. The system does not actually update any records unless the "O" character (=OK) is given.

Summing up, the update process has the following flow (cf. Figure 2):

1) Type "RUN RAMAS" to access system

2) From access format display: enter search key (1 to 7) or full name (9)

3) Select retrieved record number (if more than one record matches)

4) Display Master record information

5) Select and process instruction G/ R/ U/ E , where
"G" means "go back" (i.e. to 2)
"R" means "repeat" (i.e. 3,4,5)
"U" means "update" (follow update instructions above, record update only follows on confirmation by "O" key)
"E" means "program exit"

6) Automatic return to 5) unless instruction "E" (exit) has been given

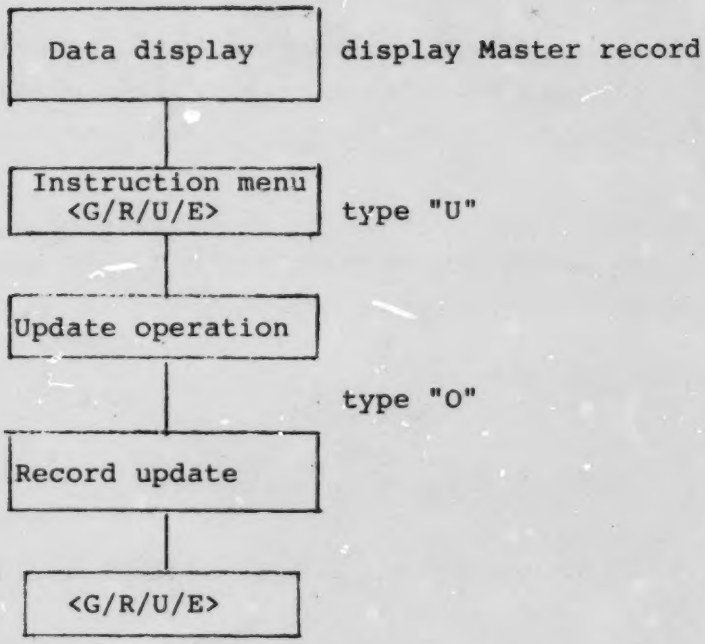


Fig 2: Updating flow diagram in Master control

1.4.3 Creating a new record

The Master Control System allows the user to create new data records on-line, after selection of the eighth point of access. After giving the type of literature, a master bibliographic record form with a new record number (which is automatically given by the computer) will be displayed on the screen for the completion of any field. The procedures hereafter are the same as in updating. The new record is created only on confirmation by instruction "O".

2. PERIODICALS CONTROL SYSTEM: INFORMATION STRUCTURE[x)]

The Periodicals Contol System manages periodicals and serials. Data presented for each issue of a journal are: arrival date (or claiming date), publication year, current volume, indication of eventual combined issues, indication of eventual missing issues, number of claims and information concerning index issue and binding. None of these data are controlled under the Master Control System.

The principal subsystems handle check-in and update operations. In addition, there are several short programs used for special applications, to print claiming letters, to issue binding notices, etc. Seven separate files are linked to the Periodicals Control System, viz:

1) Bibliographic master file
2) Periodicals file
3) Calendar file
4) Supplier address file
5) Old issues file
6) Binding file
7) Combined and missing issues file

From the programming aspect, serials control is a special case of periodicals control which is described in the following.

2.1 Data storage and representation

In view of the requirement to store date information for all issues, to save storage space, a two-byte internal string represents the 6-digit external date message (both for arrival and claiming).

---

x) In the sequel we will use the term "latest" (in quotation marks) to indicate the latest arrived issue of a periodical (serial) with respect to the natural order of publication. This means the order given by volume number, and within volume by issue number. "Latest" therefore does not refer to publication or arrival date.

Six digits are needed to express an arrival date or claim date (e.g. 821030 means 30 October 1982), but this can be reduced. If we use a 3-digit day number on annual basis instead of month and daywithin-month, and if we count the year starting form 1980, the situation will be changed a lot. For example, the string 2001 stands for "1st day of second year after 1980", i.e. 1 January 1982, and similarly 12366 stands for 31 December 1992. Furthermore, we can use a 2-digit ASCII character string (using a 128-count system instead of decimal system) to represent this shortened form of date expression. In this way the computer, for example, puts

841129 (29 November 1984) as !n,
861207 (7 December 1986) as 1E.

We add 500 to the arrival date to distinguish it from claim date. For example the computer first converts 861224 (24 December, 1986) to 6358, then to 6858 if it is an arrival date, finally to the two ASCII character string "5J". Because the maximum number that can be represented by two ASCII characters is 16384, we presently limit the acceptable entry dates within the interval 1981.01.01 to 1995.12.31. This is thought to be effective for the foreseeable future. However, changing the start year from which the system begins to count, we can of course create a different interval.

Other information for each issue includes the issue number (for which we use one ASCII character), the publication year and the claiming order number (both of which can be jointly represented by another ASCII character). The publication year (1 digit, 0-9) is used for generating claiming letters, for identification of loans requests and possibly for some statistics. The claiming order number (1 digit, 0-9) is used to select text of claiming letters and may be useful for adaptation of the delay parameter in automatic claiming and for statistical analysis of supplier's performance.

Together a 4-digit ASCII code contains these basic data for any one issue. In addition, the volume number for each issue is also an important parameter. This is dealt with in a "general" way (i.e. not displayed for each issue) by a field for "current volume" referring to the "latest" arrived issue (i.e. with the latest volume/issue number).

2.2 Check-in image and some extra parameters

In order to show the information of previous issues and to assemble the information for new arrivals, we use a check-in format with an image given in Appendix 3.2.

The information of the first five lines comes from the Master Bibliographic File. The following parameters may need comments:

a) Periodicity

This one-character code indicates the frequency or rate of publication of a journal. Together with the claim delay parameter it is used to calculate the claiming date of future issues and of older missing issues. Even for aperiodic serials it can be used to indicate an average or expected recurrency date of new items with automated alerting for review of the holdings solution.

b) Claim delay

A term of tolerance after which a claim may take place for an issue failing to come on time. The periodicity code and claiming delay parameter are defined as follows:

Publication rate	periodicity code	claim delay parameter (days)
daily	d	h (15)
twice a week	c	h (15)
once a week	w	t (21)
every second week	e	a (30)
twice a month	s	a (30)
monthly	m	f (45)
every second month	b	d (60)
once every third month	q	d (60)
three times a year	t	d (60)
twice a year	f	q (90)
yearly	a	s (180)
every second year	g	s (180)
every third year	h	s (180)
irregular	space	space

c) Number/Volume

The number of issues per volume. It is used for automatic change of the "current volume" indicator at check-in of new issue(s)

d) Index mode

This parameter indicates the way in which the index part of a volume is published. There are five index mode codes for periodicals, defined as follows:

b the index of the volume is loose
d volume-index is a part of the last issue
g index is a part of the first issue of the following volume
l index is published as a separate volume
m the periodical is a subseries and is indexed in the volume index of the main series

Obviously for journals, except those with index mode "b", when all issues of a volume have arrived, it can be reported for binding. In most cases "b", the index should be arriving with a certain issue.

Below the title field there are 69 (= 5+4+16) data fields which constitute the data record of a journal in the Periodicals file:

e) Combined Vol (No)

This data field is used to record information on combined issues, by stating volume number and combined issue numbers. For example "24:3/4" means that the third and fourth issues of Volume 24 are combined into one physical item.

f) Missing Vol(No)

This data field records eventual missing issues which, in their natural sequence precede the "latest" arrived issue by more than 11. The reason is that the 11 preceding issues along with the "latest" arrival and 4 projected ones are displayed on the screen. The last volume number in this field corresponds to the first issue for which full information is displayed on the screen (cf chapter 3).

g) Current volume

This field indicates the volume number corresponding to the "latest" arrived issue. It will be automatically changed, as determined by the No/Vol parameter.

h) Future issue year (of publication)

This field gives the expected publication year of the last issue for which full information is displayed, i.e. the fourth issue to follow the "latest" arrival. It can be changed both automatically and manually. At present, the change of year of publication has to be entered manually, but then is recorded automatically until the next manual change. When the first issue of a volume appears initially among the four future issues, the system asks for eventual change of year in the "year" field. At present, the publication year is displayed only for the four future issues, but is kept in record for all other issues. It will therefore appear correctly in claiming letters.

i) No, arrival date, claim date and CNo Yr

As mentioned before, one issue has four data features recorded by a 4-digit ASCII code. These are: issue number, arrival (or claim) date, claim order number and publication year. In the check-in image there are a total of 16 groups of corresponding data fields. This means that full information for a maximum of 16 issues can be displayed on the screen. The first 12 groups are used for issues up to the "latest" arrived one (in the natural order), and the last 4 groups for the coming 4 issues with calculated future claim dates.

j) A special comments field (27 characters) is added to the check-in sequence

It can be used to alert on advance reservations of expected issues. In the context of a circulation system this information can be utilized to generate notifications automatically.

3. PERIODICALS CONTROL SYSTEM: CHECK-IN

When a given ISSN-number (or record number) is keyed in, first of all the system displays a check-in image for periodicals (Appendix 3.2), and then fills in information on the corresponding title, holdings, issues per volume, claiming mode, index mode, etc, all of which come from the Bibliographic Master file. Next, the system displays information on the previous arrived or claimed 12 issues, up till the "latest" arrival, and the coming 4 issues (with issue number and related future claim dates calculated automatically, based on the periodicity code, claiming delay parameter and the number of issues per volume recorded in the Master file). The cursor always initially points to the arrival date field of the issue expected to arrive next (i.e. following the "latest" issue recorded as arrived).

In order to enter an arrival date of a new issue, one either inputs the 6-digit date from the keyboard (in the order year-month-day, e.g. 821015, which means 15 October 1982) or uses default present-day-mode, i.e. pressing the "0" key on the keypad plus terminator. If there is more than one issue to be checked-in, one must use the "TAB" key instead of the "RETURN" key as a terminator. Using the "RETURN" key always means to stop check-in. Unless the default mode is used, the system will check keyed-in date and reject illegal date (e.g. 823212 or 830229). In such a case, it is necessary to re-enter a correct date.

The system is also able to deal with the case of combined issues. In this case, to enter combined issues, first locate the cursor at the lowest relevant arrival field by the TAB key or "backspace" key, then key-in "2" plus TAB terminator, and then key-in "0" plus a terminator. After that, information on volume number and combined issue number will appear in Combined Vol/No field, in addition to earlier information.

When there are 12 arrived or claimed, and the 4 predicted issues displayed on the screen, the first old issue will move out of display at check-in of a new issue. However, the information of the removed issue will be kept in the "old issues file". Minimum information on such older issues, e.g. on combined issues or unsatisfied claims ("missing issues") is still displayed in special fields so that simple updating concerning late arrival of missing issues is possible. When this information exceeds one line, the older information will be transferred to a special file and the new line on the screen will be flagged by a " * " mark. A special program will be needed to see those old informations. This will be solved in the context of the circulation/loans system.

The last volume number appearing in the "missing issues" field always corresponds to the volume number of the first issue in the full display, and the number in the "current volume" field corresponds to the "latest" arrived issue. Both volume numbers are changed automatically as new issues are checked in.

To sum up, for each issue the following information is recorded:

a) Issue number (generated by the system, based on No/Vol-parameter)
b) Next claiming date (generated by the system) or arrival date (given at check-in)
c) Number of claims dispatched (generated automatically, see chapter 4)
d) Year of publication (given manually on appearance of 1st issue of a new volume among expected issues)

For check-in of a new issue, the simplest mode of access to the periodical record is to key-in the ISSN number (given on the issue). The display gives bibliographic information for control, and the cursor automatically points to the first new issue expected to arrive. To enter present date message, one can use either the default mode ("0" on the keypad) or key-in the 6-digit date on the keyboard. One can input any formally acceptable date in the 6-digit way as approrpiate. Each date entry must be accompanied by pressing a terminator key.

There are three terminators that can be used:

1) "TAB". It is a terminator for date-entry and moves the cursor forward. It has no ending check-in function
2) "Backspace". It is a terminator for date-entry and moves the cursor backward
3) "Return". Stop check-in

In some cases, the journal has no ISSN number or the ISSN is not given on the journal issue or is not easily available. Here it is necessary to find the record number in the Master file by accessing through title (or title key), and then to restart the system for check-in. (This procedure may be improved so that the system automatically transfers to the related record in the Periodicals file.)

The check-in program has the following procedure (cf. Figure 3):

1) Type "RUN PM", thereby calling Periodicals Management System.
2) Give the ISSN-number or record number.
3) Display bibliographic information given by periodicals file.
4) The cursor automatically moves to "expected arrival" field: key-in date, if "expected" issue is handled, else move the cursor to desired field and key-in date. A special procedure is followed in case of check-in of combined issues.
5) Press "RETURN" key to stop present check-in.
6) Press "RETURN" key again to start next journal check-in or type "E" for exit.

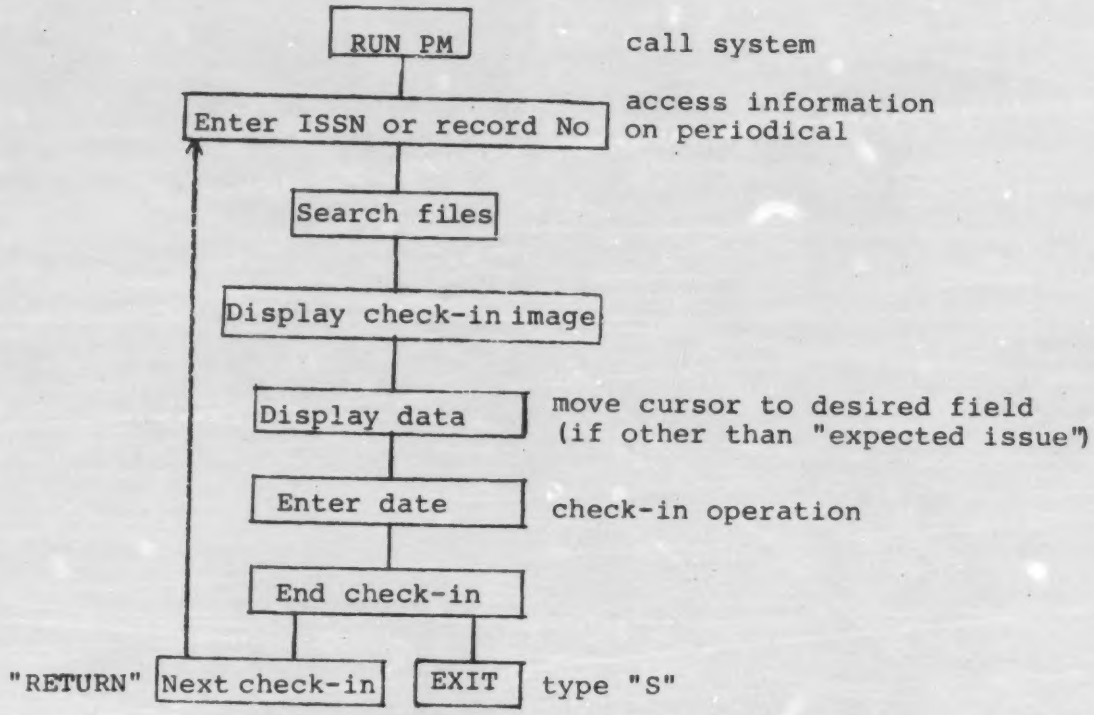


Fig 3: Flow diagram for check-in in Periodicals control

4. PERIODICALS CONTROL SYSTEM: CLAIMING

As was described before, a claiming date is created for each issue when it first appears on the display among the four expected issues to follow the "latest" arrival. For a new periodical record, claiming dates for the four following issues are created upon the first check-in. (The first issue can be regarded as a regular acquisition, the control of which falls under the acquisition procedure and its claiming routines.) As new issues are checked in, the band of 16 fully displayed information sets is filled up and subsequently moves one step for each new check-in. As long as an issue has not arrived, its calculated claiming date will be associated with the issue number. On check-in of the arrived issue, the claiming date will be cancelled and the arrival date inserted.

The system will control each claiming date by a calendar file, which is updated simultaneously with the creation of claiming dates in the periodicals file. The calender file is independent of the display limitations to 16 issues. If an issue has not arrived on the recorded claiming date, running a special application program, a claiming letter stating journal name, volume (publication year) and issue to be claimed, and supplier's address will be printed out on the line printer.

On the event of every claiming, a new claiming date with a period given by the delay parameter will occur in the periodicals record and also in the calendar file. Further, the parameter stating the claiming order number will be increased by one.

It should be noted that the claiming dates can be changed manually even in the automatic claiming mode in order to cater for special circumstances of publication. In the non-automatic mode, apart from manual entry of claiming dates, the claiming procedure will be the same as described.

In case the claiming letter routine has not been activated for some time there is a special feature which guarantees that new claiming dates will always refer to the future even if the normal calculation would yield a past

date. This is to avoid several letters to be sent out within a short period for the same claims.

What actually happens is a complex retrieving, reordering and updating process, where the program has to:

a) scan the Calendar File,
b) get the record number of overdue issues,
c) for each number picked up, access the periodicals file, create new dates for next claim, and substitute new claim date, and step up claim order number,
d) access the Master Bibliographic File, get the title of the journal and supplier code,
e) get the full address of supplier from the Address File,
f) finally print out a claim letter as shown in Figure 4. All claims related to a certain journal title will be listed in the same claim letter.

Under MCR-status, the order "@ LETTER" will cause all pending claiming letters to be printed out on the line printer.

ROYAL INSTITUTE OF TECHNOLOGY LIBRARY
Acquisition Department
S-100 44 STOCKHOLM, Sweden

Stockholm (date)
(Address of supplier)

CLAIM FOR MISSING ISSUES
Ref.: Our Subscription for: ..... (Title)
Missing issue(s): Vol No (Year), Issue No; Vol No (Year), Issue No; .....

Please, supply and/or inform us of action taken.

Your reply:

Fig 4: Claim Letter Format

The claiming routine is roughly summarized in Figure 5.

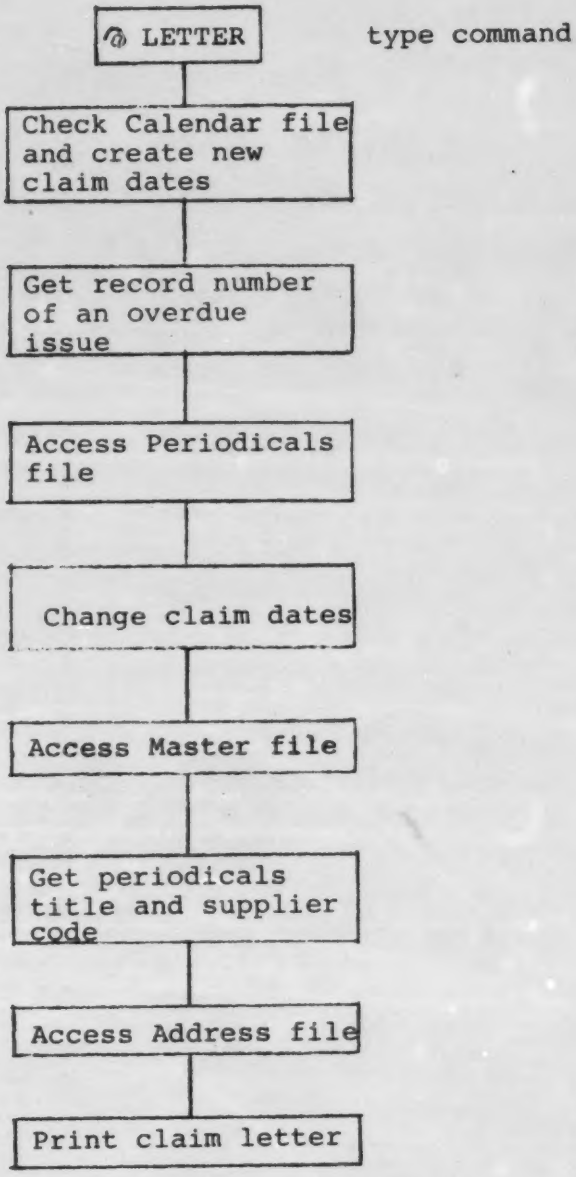


Fig 5: Automated Claim Flow Diagram

5. BINDING

While running the check-in procedure, the system will also control the registration conditions of each volume. This is done by means of the index mode code (see 2.2.d) and a binding file. In most cases the index to a volume is mode available in a regular fashion. Only the case with index code "b" is irregular in the sense that the index of each volume is loose and it is not clearly stated with which issue it will arrive . But "statistically", from experience of several previous volumes, one may roughly tell that the index of the volume should probably arrive with a certain issue. It is necessary to enter that expected issue number beforehand on a special display to be used as an indicator for claiming. This is done when the record of a journal in the periodicals file is created. This parameter will be stored in the Binding file. Thereafter, to get an index message, when the "RETURN" key in the check-in procedure has been pressed, the system will sometimes display a registration image at the last line of the screen. This image includes claiming information for the index part, related to the expected issue of the periodical. This image looks as follows:

IDX come (Y/N). Expected issue... Claim date.... Claim delay .... Claim No....

In check-in, this image is only displayed when the expected issue number for index delivery comes to within 4 issues of actual check-in and does not disappear until index issue has arrived. The only thing one needs to do when an expected index issue has arrived is entering "Y" where the cursor stays, before pressing the "RETURN" key. Otherwise one only has to press the "RETURN" key because the default letter in "IDX come (Y/N)" field is "N" which means that the expected index issue has not come. Apart from the case with index code "b", there is no image display at all after the check-in procedure of any journal, since the binding notice procedure is then fully automated. When a complete volume (issue and index part) is recorded as arrived, a "can be bound " message and "check date" (which refers to the related check-in date) will be recorded in the Binding file, and the claim date will be erased. Based on the "can be bound " message, the system will print out a binding notice in cases where binding is requested.

To do this under MCR status, the order "@ BINDING" running in the "notice" mode will cause all pending binding notices to be printed out on the line-printer. The binding notice will inform on the journal volumes which can be processed for binding and where the index part is to be found (cf. Fig 6).

Signum	Vol	Index mode	Check date	Journal title
xxxxxxxx	xx	Loose, next x	xx-xxx	xxxxx xxxxx
xxxxxxxx	xx	Next 1st	xx-xxx	xxxxx xxxxx
xxxxxxxx	xx	Last issue	xx-xxx	xxxxx xxxxx
xxxxxxxx	xx	Subseries	xx-xxx	xxxxx xxxxx
.		.	.	.
.		.	.	.
.		.	.	.

Fig 6: Binding Notices Format

When making the binding notice, the binding program accesses the Master file and writes a 10 letter message, like "821120V020" into the binding field of the record concerned, which in this case means that volume 20 has been bound and completed volume is ready on 20th November 1982. The same message will be written into the comments field of the related record in the Periodicals file. Either from the Master Control System or from the Check-in System one may see which volumes are ready for binding or on-binding. After issuing the binding notice in the form illustrated in Fig 6, all the states of "can be bound" will be changed into "on-binding" states. To avoid duplication, no more binding notices in this form can be issued for these particular volumes. If information on all volumes ready for binding or on-binding is needed, one can run the Binding program in the "check" mode, searching the binding file and print a list with general layout exemplified in Fig 7. The binding date in Fig 7 refers to the date when the binding notice information was issued.

Record number	Signum	Vol	Binding date
1012	2eÖ7720	18	821024
3100	6eÖ701	13	821024
3102	6eÖ703	41	821101
3127	5eÖ6781	7	821101
.	.	.	.
.	.	.	.
.	.	.	.

Fig 7: List of volumes ready for binding (or on-binding)

When a volume has returned back from binding, it is necessary to cancel the message in both the Master file and the Periodicals file, since otherwise the bound volume is still kept in on-binding status. To do this, first of all, one has to run the Binding program and give a cancel command. This gives rise to printout of a list as shown in Fig 7. Using this list one can line by line, check which volumes are ready. One then enters only the serial numbers to be cancelled, in response to question asked by the system. A "RETURN" key follows each record number. The process does not stop until "RETURN" key is pressed without any record number (Fig 8).

```
② BINDING
Run Binding
Command: notice/check/cancel (1/2/3):3
Cancelled number: xxxx CR
Cancelled number: xxxx CR
Cancelled number: CR
End of entry
Number of cancellations = xx
```

Fig 8: Procedure of cancelling binding notices

On termination of the cancelling operation, the system shows the total number of entries to be cancelled and begins to erase the corresponding binding message in the Binding field of the Master file record and the Comments field of the Periodicals file record concerned. At the same time information on the bound volume will appear in the Holdings field of the same Master file record.

In case the index of the volume is loose, a special claiming procedure is needed, since the issue which is expected to "carry" the index of the previous volume may arrive without that index. When this situation occurs the system will print out a claiming letter, based on the parameters in the binding file such as Claim Date and Claim Delay, to inform the supplier in a similar way as it does claiming for normal issues. This special function is not operative at the moment.

Acknowledgements

The author wishes to thank my supervisor, Dr. S. Schwarz, director of RITL, for insisting that I should work on this important and challenging problem, for providing a stimulating environment for R&D activities, and for numerous discussions throughout the entire design process. Further I appreciate discussions and helpful suggestions by Bo Johansson, Stefan Svanström, Stefan Ahnström, Arne Nilsson, Eva Abrahamsson, Gunnel Werme and many other colleagues at RITL.

APPENDIX 1 :

Bibliographic Master File Data Structure

Field name	Display length	Data code & name in periodicals	in monograghs etc		bit No
F1(A1)	7	Record number	Record number	KF1(ASCII)	3 (0-2)
(non-display)		Title key	Title key	KTL	6 (3-8)
(non-display)		(---)*	Originator key	KAU	6 (9-14)
(non-display)		Conference name key		KCO	6 (15-20)
F2(A2)	13	ISSN	ISBN/NBN	KF2	13 (21-33)
F3(A3)	12	Signum	Signum	KF3	12 (34-35)
F4(A4)	19	(not exist)	Classification	KF4(ASCII)	7 (46-52)
F5(A5)	1	Type of Literature		KF5	1 (53)
		(total length of KEYs: 54)			

(following fields are variable)

				(Pointer No	& length)
F6(A6)	1		Location RITL	0	(64)
F7(A7)	1	Seperate catalog	(---)		
F8(A8)	6	(---)	Order date		
F9(A9)	6	(---)	Arrival date		
F10(A10)	50		Holding	1	(240)
F11(A11)	80		Conference name	2	(240)
F12(A12)	80		Title	3	(71)
F13(A13)	71	Library	Originator	4	(71)
F14(A14)	71		Corporate body	5	(62)
F15(A15)	2		Country		
F16(A16)	2	(---)	Ed.Version		
F17(A17)	58		Impressum	6	(240)
F18(A18)	80		Comments	7	(146)
F19(A19)	66	Variant title	Serial title		
F20(A20)	80	Variant title	Serial title		
F21(A21)	3		Supplier code	8	(21)
F22(A22)	1		Supply mode		
F23(A23)	1	(---)	Standing order		
F24(A24)	1		Periodicity		
F25(A25)	2		Number/vol		
F26(A26)	1	Index mode	(---)		
F27(A27)	1		Claim mode		
F28(A28)	1		Claim delay		
F29(A29)	10	Binding	(---)		
F30(A30)	3		Currency	9	(103)
F31(A31)	10		Missive number		
F32(A32)	67		Price		
F33(A33)	1		Duplicated signum/class. mark		(1)
POINT	1(not data)	Select pointer			

*) (---) indicates that the field does not exist

APPENDIX 2:

A 2.1 Bibliographic Image of Periodical

No: ..... ISSN ..... Signum ..... Classification.....

Type . Location RITL . Seperate catalog .

Holding .....

Conference name :

.....

.....

Title:

.....

.....

Variant title: .....

.....

Corporate body: .....

Library: .....

C . Ed. .. Impressum .....

Comments:

.....

.....

Supplier code ... Supply mode . Period . N/vol .. Idx.md . Cl.md . Cl.delay .

Binding ..... Currency ... Missive number..... Price

.....Dup.S/Cl..

A 2.2 Bibliographic Image of Monograph etc.

No: .....ISBN/NBN ..... Signum ..... Classiffication .....

Type . Location RITL . Order date ..... Arrival date .....

Holding .....

Conference name:

.....

.....

Oraginator(Author, Editor, Compiler, Coauthor):

.....

Title:

.....

.....

Serial title .....

.....

C .. Ed. .. Impressum .....

Comments:

.....

.....

Supplier code ... Supply mode . Standing order . Period . N/vol ..

Claim md . Claim delay . Currency ... Missive number..... Price

..... Dup.S/Cl..

APPENDIX 3:

A 3.1 ACCESS IMAGE

```
Access point          ..

0    Exit
1    Record number    .......
2    Title  key       ......
3    Author key       ......
4    Conferance name key  ......
5    ISSN/ISBN/NBN    ............
6    Signum           ...........
7    Classification   ...................
8    Enter new record .
9    Full name of     .
     a  Periodical,Book
     b  Author
     c  Conference

.....................................................................
.....................................................................
.....................................................  ..............
```

A 3.2 Check-in Image for Periodicals

```
No: .......  ISSN  ..........  Signum ..........Loc.RITL . Separate catalog..

Period . N/Vol .. Index mode ......  Claim mode . Claim delay ...

Title:
.....................................................................
.....................................................  ..............

Combined Vol(No)  .....  ...........................................
Missing  Vol(No)  ...........................................

Current volume  ...  Future issue year (..)
         No  .. No  .. No  .. No  .. No  .. No  .. No  .. No  .. No  ..
Arrived  ..... ..... ..... ..... .  .. ..... ..... ..... ... ..
Claim    ..... ..... ..... ..... ..... ..... ..  . ..... .....
CNo.Yr.  ..... ..... ..... ..... .  .. ..... ..... ..... .....

         No  .. No  .. No  .. No  .. No  .. No  ..     Comments
Arrived  ..... ..... ..... ..... ..... .....  ..........................
Claim    ..... ..... ..... ..... ..... .....
CNo.Yr.  ..... ..... ..... ..... ..... .....

Index ready(Y/N) . Expected issue .. Claim date ..... Claim delay ... Claim No.
```

A BRIEF SURVEY OF THE RAMAS PROGRAMS

The Master control, Periodicals control subsystems and some subroutines for special applications, which have been developed and implemented in RAMAS, consist of the following sets of subprograms:*)

1. RAMAS: Master control subsystem

   LX, INIT, NOASC, NODEC, ACCES, DISPLY, UPDATE, REWRIT, WRIT1, WRIT2, WRIT20, WRIT3, ENAR,TKEY,SELECT

2. PM: Periodicals control subsystem

   PM, GETDAT, CHEIN, CHECK1, CHECK2, UPDVAL, DISMOV, DISPLP, DATEC, CHECK3, ACCESM, DISPM, RENEW, CHOICE, CALEND, BIND1, BIND2

3. CL: Claiming letter module

   CL, ACCES, GETDAY, CHECK, RENEW, CALEND

4. BIND: Binding notice control module

5. TX: LIST TECH translation module

---

*) The complete listings of all programs described here are available in a limited edition of the report:
Lian Ya-chun: RAMAS: THE RITL AUTOMATED MANAGEMENT SYSTEM, Computer programs for Master control and Periodicals control (Report TRITA-LIB-4081, January 1983)

1. RAMAS: Master Control Subsystem

1.1	LX	The main program of the Master Control subsystem. It calls 11 subroutines and controls the program flow.
1.2	INIT	The initializing subroutine of RAMAS. Its functions include to open the bibliographic main file - BIBL-MAIN.FILE, to display the access image, to provide access to the system and to receive data given by the user.
1.3	NOASC	The data conversion subroutine. It converts a decimal expression of numeric data into its ASC11 representation.
1.4	NODEC	The data conversion subroutine. It converts a ASC11 representation of data into decimal form.
1.5	ACCESS	The access subroutine. It fetches the data from the main file in the way supported by the initialized subroutine and distributes the data into the matrix elements of AA$
1.6	DISPLY	The display subroutine. It calls for the bibliographic image display and fills the data of the matrix AA$ into the relevant data fields of the bibliographic image on the terminal screen.
1.7	UPDATE	The update subroutine. It is used for updating the information in any data field except the record number.
1.8	REWRIT	The pre-write subroutine. It reorganizes the record data after the updating operation has been completed. REWRIT calls the following three subroutines: WRIT1, WRIT2, WRIT20.
1.9	WRIT1	The variable field recombination subroutine. It combines data from the elements of AA$ into the data components.

1.10 WRIT2 The variable counting subroutine. It creates the pointer number and the counter value related to each component.

1.11 WRIT20 The variable accumulation subroutine. It puts all variable components together.

1.12 WRIT3 The write subroutine. It writes a data record into the bibliographic main file and displays the record length in the last line of the screen.

1.13 ENAR The subroutine for creating new records. It gives a new, unique record number and selectively displays the bibliographic image.

1.14 TKEY The subroutine for creating keys. Its functions include creating the title key, conference name key and author name key.

1.15 SELECT The selection subroutine. It is used for picking up the record number when serveral records with the same access key are hit in the same search.

2. PM: Periodicals control subsystem

2.1 PM The main program of the periodicals control subsystem. It initializes the form system, calls nine seperated subroutines and controls the program flow.

2.2 CHOICE The subroutine to select the way the user is going to access the data base. It opens the bibliographic main file and performs digital conversion.

2.3 ACCESM The subroutine to access the Master file. It fetches only the information the PM system wants from the Master file. This includes "periodicity", "number per volume", "index mode", "claim mode", "claim delay", "title" , etc.

2.4 DISPM The subroutine to display information from the Master file on the check-in image for periodicals.

2.5 GETDAT The subroutine to access the periodicals file. Its functions include: to open the periodicals file, to give access to relevant records, to distribute and display the data in corresponding data fields in the display image.

2.6 CHEIN The check-in subroutine. It locates the expected "next arrival" and calls the following second level subroutines: DATEC, CHECK1, UPDVAL, DISMOV, CHECK2, DISPLP, CHECK3.

2.7 DATEC The subroutine dealing with the default present date and the combined issues.

2.8 CHECK1 The check subroutine. It checks the validity of a date when it is entered without using the default key.

2.9 UPDVAL The subroutine updating the volume number in the missing issues field. If information on a non-arrived issue has to move out of the image, UPDVAL will check its issue number and also update the volume number in the Missing Issues field when a last issue of a volume is moved.

2.10 DISMOV The dismovement subroutine. It performs five functions: 1st, dismoving the data concerning each issue from right to left (once the 16-squares area is filled); 2nd, calculating the future four issue numbers; 3rd, check the year of publication when the 1st issue of a new volume appears; 4th, to assign the year of publication, 5th, to update the current volume number.

2.11 CHECK2 The subroutine used for assigning a claim date to each of the four future issues, based on the date of the "latest" arrival.

2.12 DISPLP The display subroutine. It displays the new information after dismoving and calculating. If the contents in the missing issues field or combined issues field overflow, DISPLP will store the old information elsewhere.

2.13 CHECK3 The subroutine to key-in claim date manually.

2.14 RENEW The re-write subroutine. It reorganizes the whole information after the check-in procedure, and transfers it into the periodicals file (update operation)

2.15 CALEND The calendar subroutine. The functions of CALEND include updating the claim date (calendar) file according to the check-in information and creating a new over-flow record if the number of moved issues exceeds 46.

2.16 BIND1 The subroutine used for checking all issues within the last volume. It is obviously a necessary condition for binding that all issues in one volume have arrived.

2.17 BIND2 The index control subroutine. Considering the index mode parameter of the periodical, and the condition that all issues of one volume should have arrived, BIND2 controls the check-in procedure of the index issue and manages the binding file.

3. CL: Claiming letters module

3.1 CL The main program used for issuing formal claiming letters. It must access the bibliographic main file, the calendar file and the periodicals file at the same time. Three subprograms have to be called: ACCES, GETDAY, CHECK. Claim messages are stored in an interim file. CL will print out all the claiming letters on the lineprinter following a search for overdue issues in the calendar file.

3.2 ACCES The subroutine to access to the bibliographic main file. It is the same as the subroutine described in 1.5.

3.3 GETDAY The subroutine to access the periodicals file. Its functions include the digital conversion which translates an ASC11 representation of the data into decimal form.

3.4 CHECK The claim accounting subroutine. It picks up the overdue issue, its year of publication and its volume number, and calculates the new claim date. It also reorganizes the ASC11 representation of the date, and finally, rewrites the new data into the periodicals file and the calendar file.

4. BIND: Binding notice control module

BIND It performs three functions related to the binding procedure: NOTICE, CHECK, CANCEL.

The function of NOTICE is to print out a notice on journal volumes that are ready for binding. It also puts a short binding message into the bibliographic main file and the periodicals file.

The function of CHECK is to print out a list of all journal volumes that have been declared ready for binding and where these notices have not yet been cancelled (on delivery of bound volumes or due to change of binding priorities).

CANCEL is used to eliminate the short message about Binding in the Master file record and in the Periodicals file record. It simultaneously updates the binding file so that the corresponding volume will not appear on subsequent lists.

5. TX: LIST TECH translation module

TX This module is used for transferring data from the LIST TECH file (the Swedish Union Catalog of S&T Serials and Periodicals) into the Master file in the RAMAS system. The relationship of the data fields in both systems is listed as follows:

No	LIST TECH	DATA NAME	RAMAS
1	020	Signum	A3
2	030	Private library	A14
3	040	Public library	A13
4	080	Impressum	A17
5	110	ISSN	A2
6	200	Title	A12, A19, A20
7	830	Periodicity	A18
8	850	Separate cataloguing	A7
9	970	Location at RITL	A6
10	980	Holding	A10
11	810	Main record mark	-

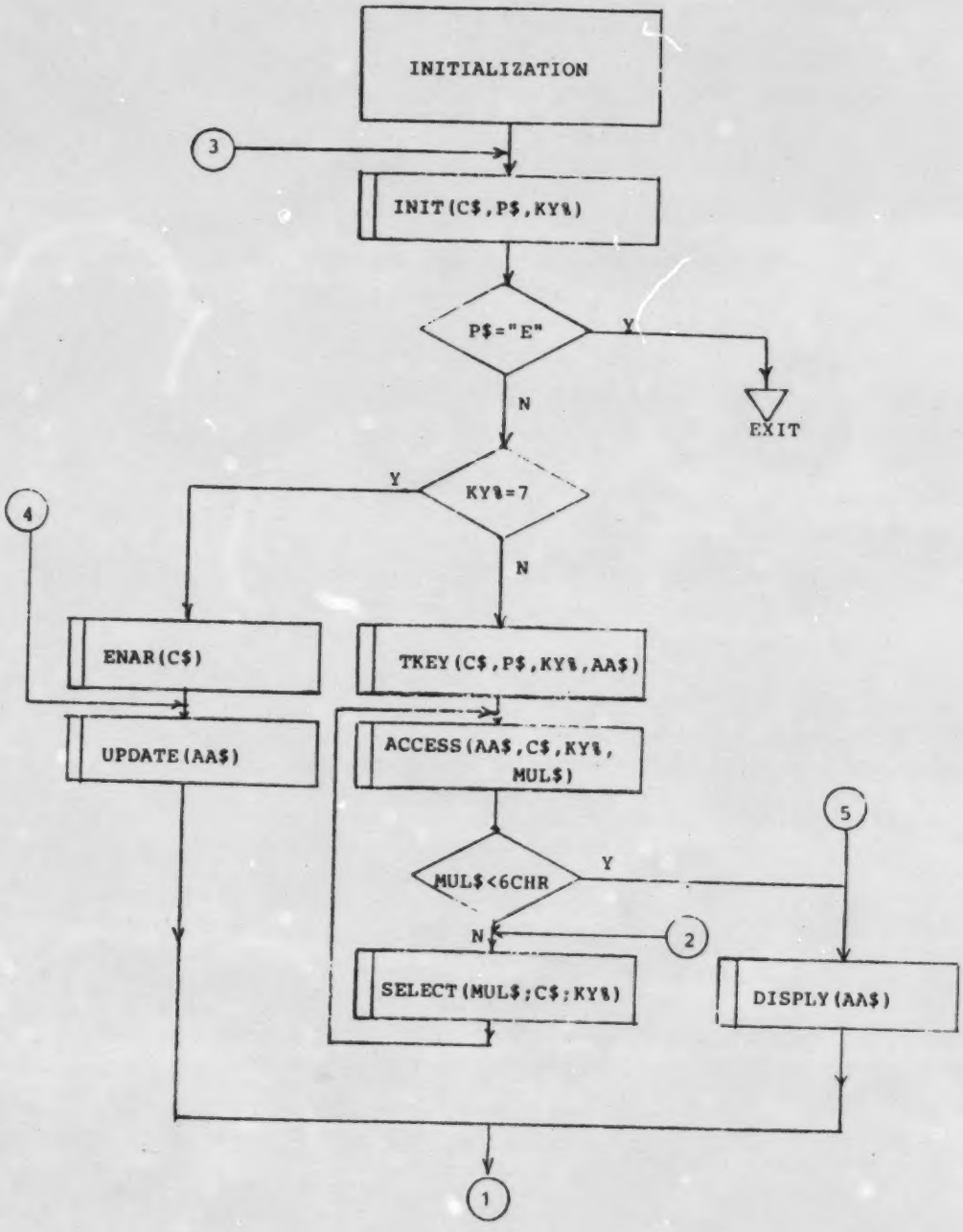
FLOW DIAGRAMS

TABLE OF CONTENTS

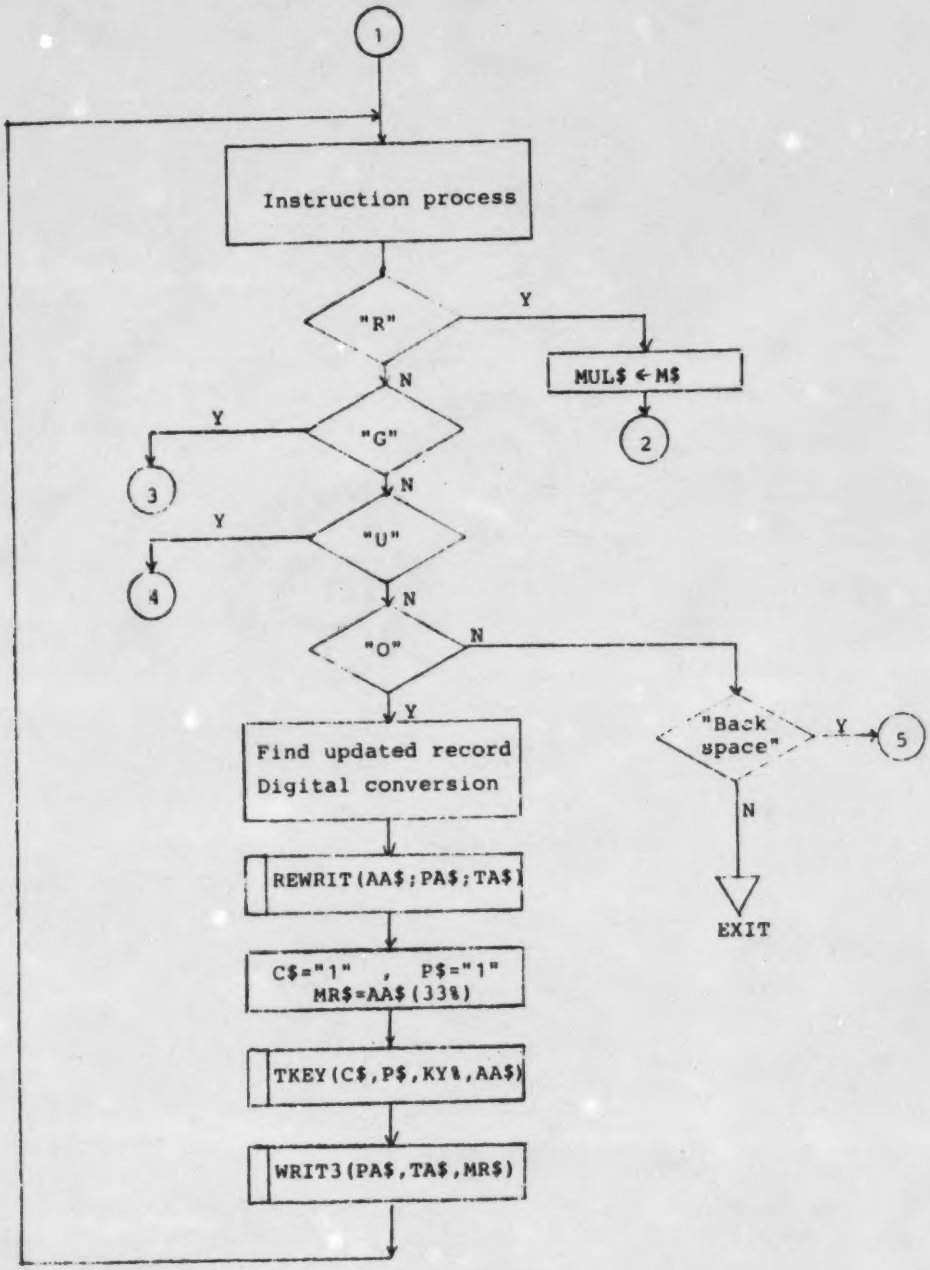
1. RAMAS Master Control Subsystem
   a) Main Program 51
   b) Subprogram INIT 53
   c) Subprogram TKEY 55
   d) Subprogram ENAR 57
   e) Subprogram ACCES 58
   f) Subprogram SELECT 59
   g) Subroutine (950) 60
   h) Subprogram UPDATE 61
   i) Subprogram REWRIT 63
   j) Subprogram WRIT 2 64
   k) Subprogram WRIT 3 65

2. RAMAS Periodicals Control Subsystem
   a) Main Program 66
   b) Subprogram CHOICE 68
   c) Subprogram ACCESM 69
   d) Subroutine GOSUB 850 69
   e) Subprogram DISPM 70
   f) Subprogram GETDAT 71
   g) Subprogram CHEIN 73
   h) Subprogram DATEC 74
   i) Subprogram UPDVAL 75
   j) Subprogram DISMOV 78
   k) Subprogram DISPLP 80
   l) Subprogram CHECK 1 82
   m) Subprogram CHECK 2 83
   n) Subprogram CHECK 3 84
   o) Subprogram RENEW 85
   p) Subprogram CALEND 86
   q) Subprogram BIND 1 87
   r) Subprogram BIND 2 88

RAMAS MAIN PROGRAM

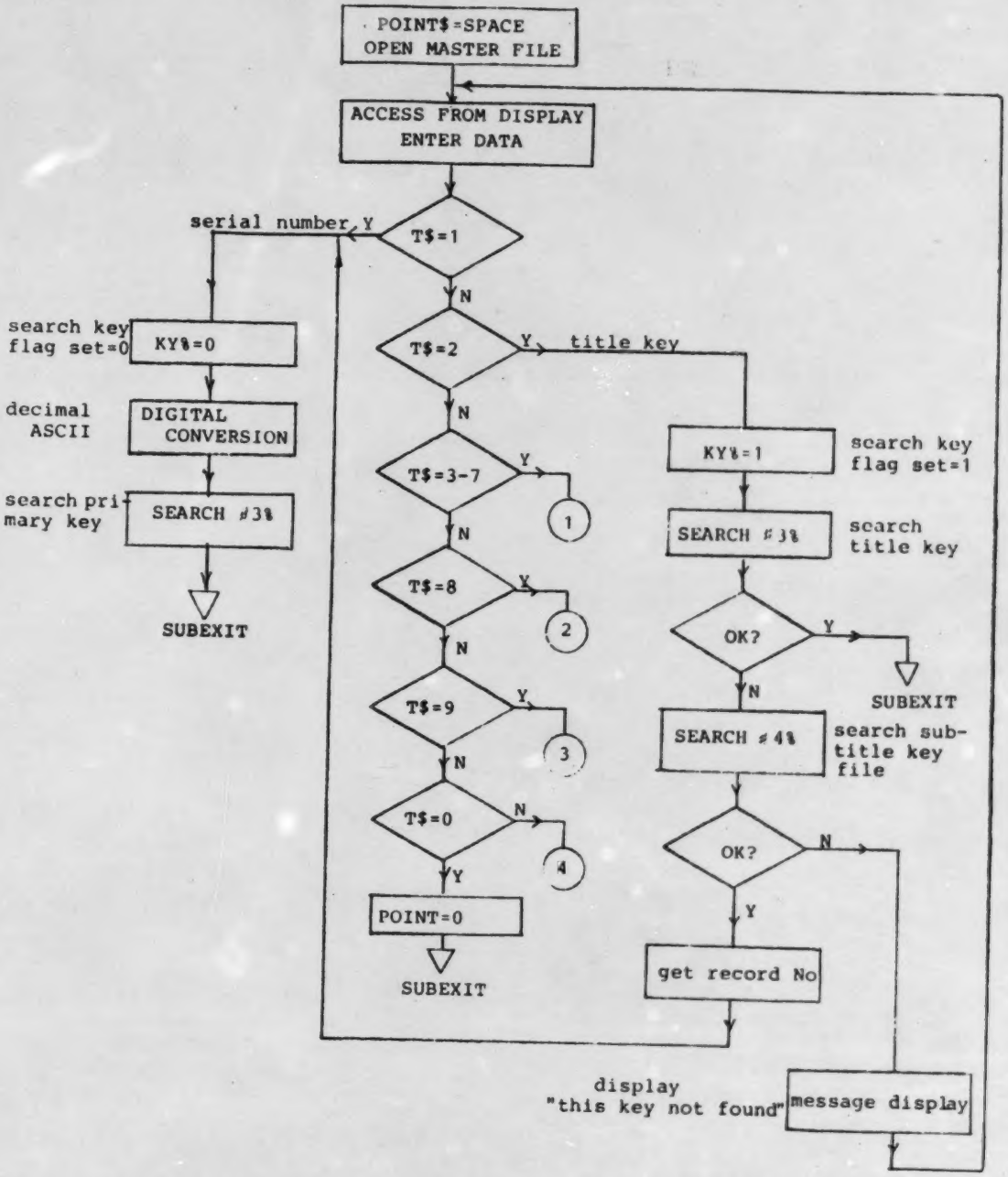


RAMAS MAIN PROGRAM (continued)

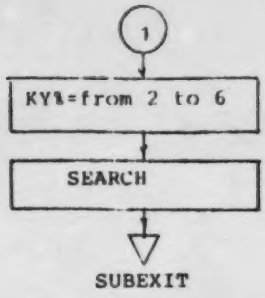


SUBPROGRAM INIT

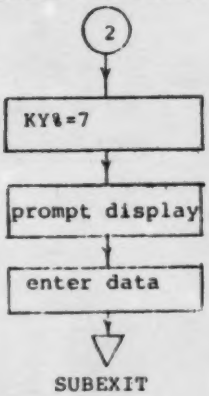
PROGRAM NAME: INIT (C$, POINT$, KY%)



SUBPROGRAM INIT (continued)

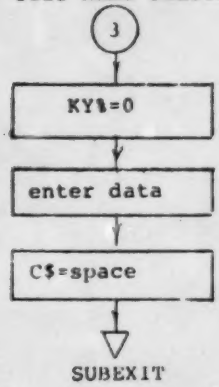


create new record



enter lit.type
P/M/C/S/V

full name search

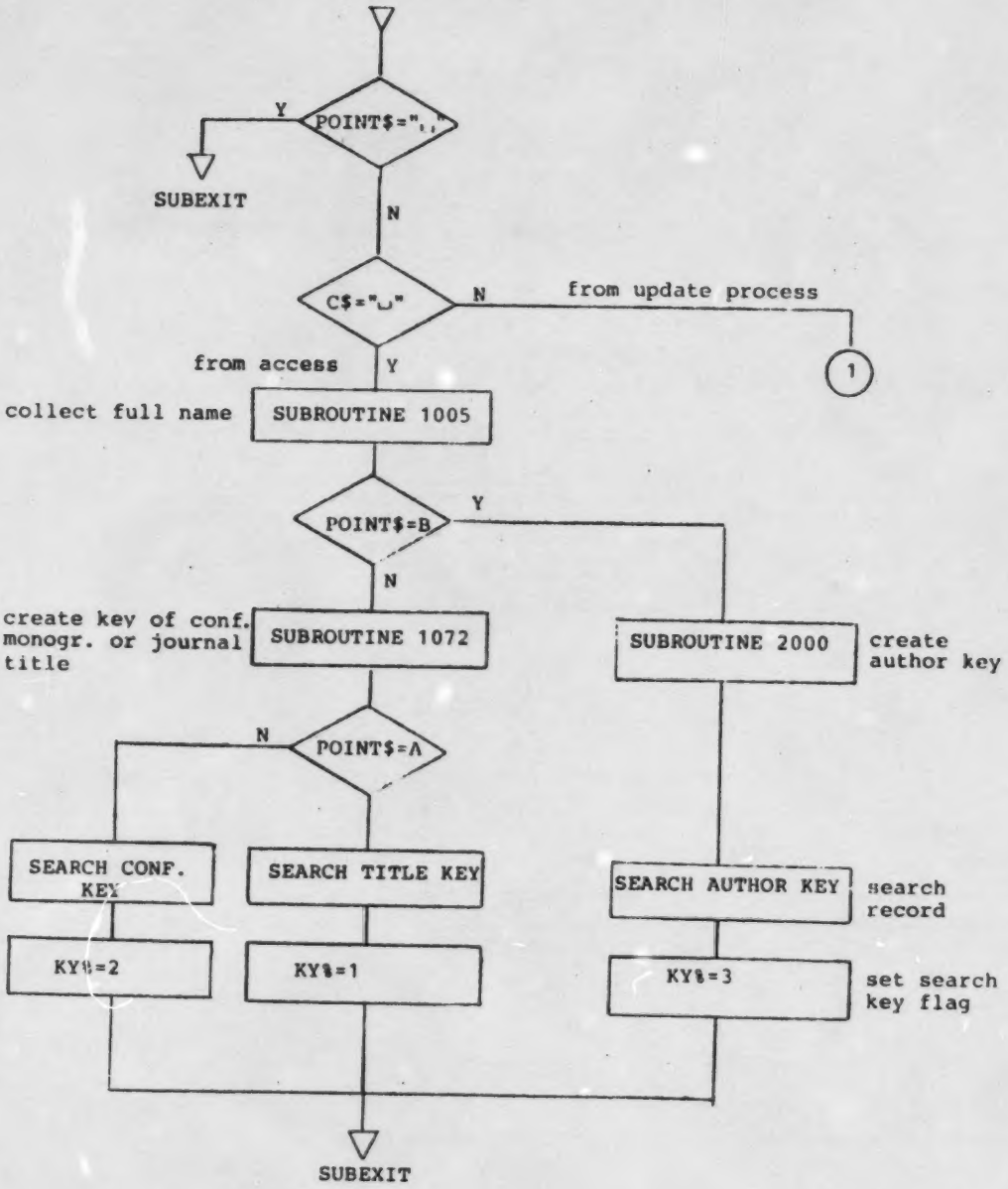


POINT$=a,b or c

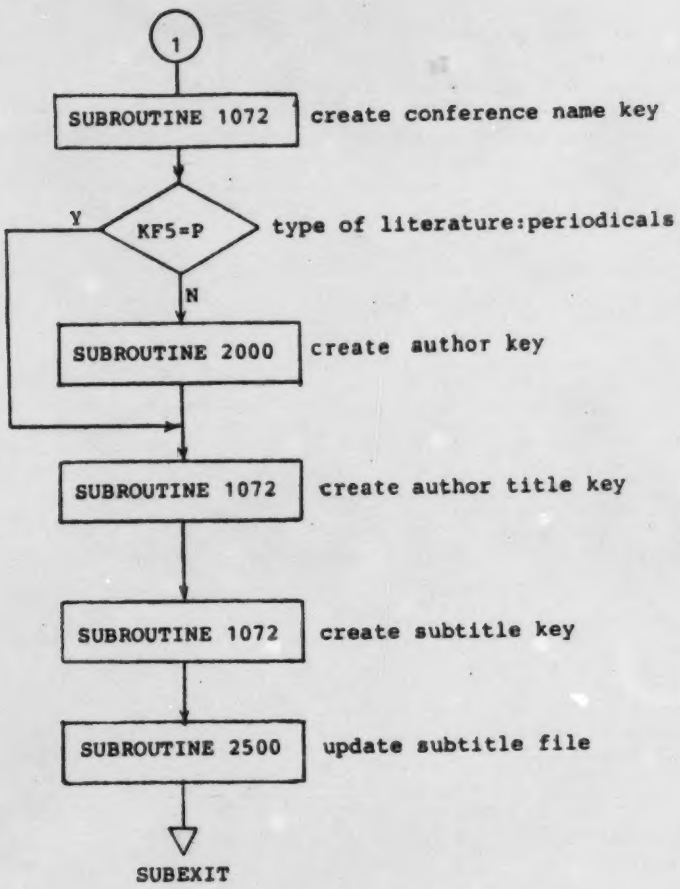
Only in full name search
POINT$/space and C$=space

SUBPROGRAM TKEY

PROGRAM NAME: TKEY (C$, POINT$, KY%, AA$(,))

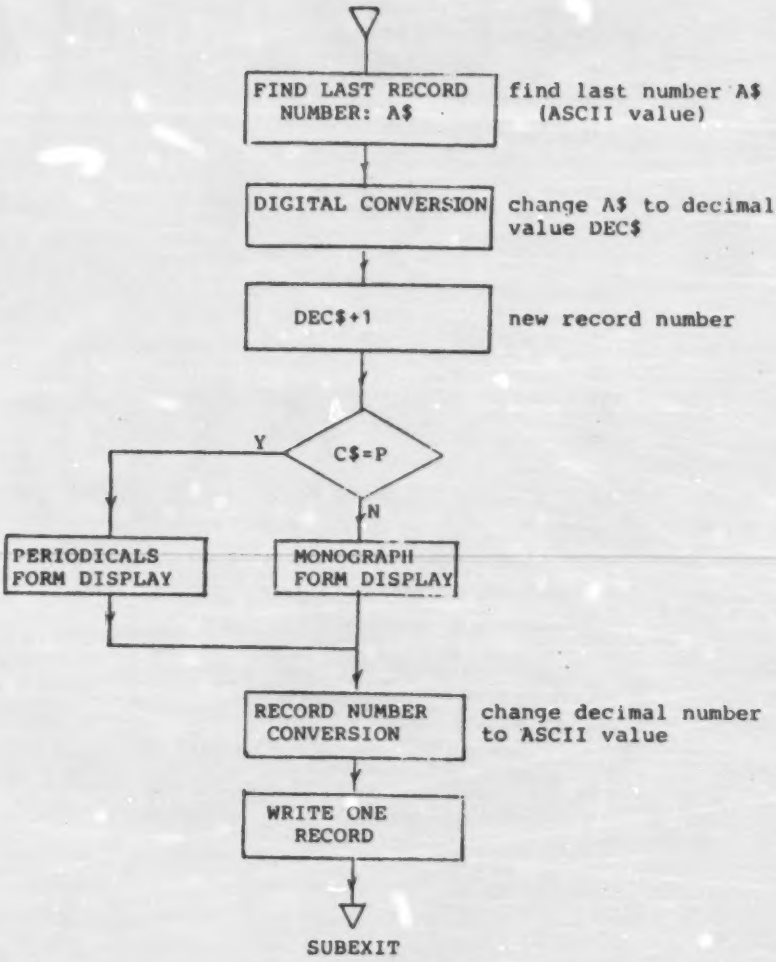


SUBPROGRAM TKEY (continued)



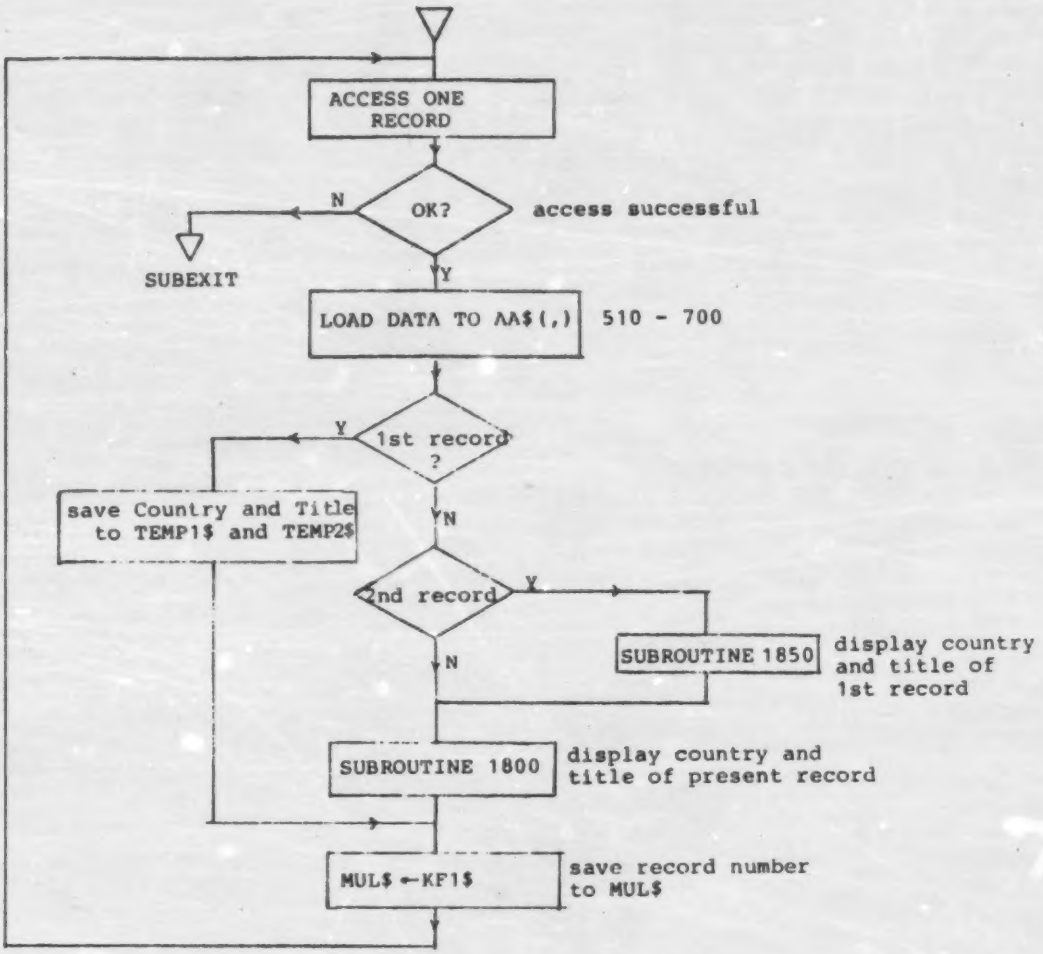
SUBPROGRAM ENAR

PROGRAM NAME: ENAR (C$)



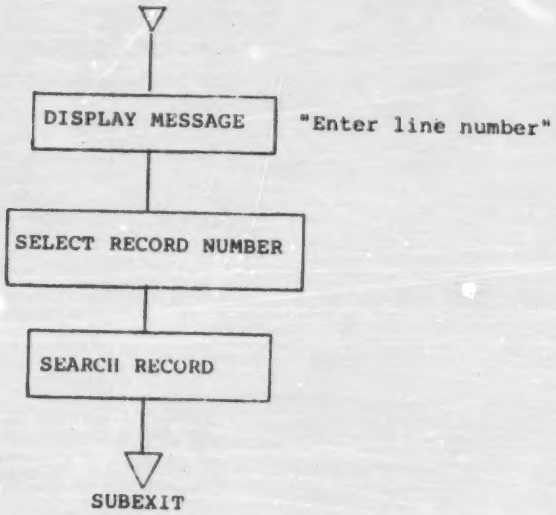
SUBPROGRAM ACCES

PROGRAM NAME: ACCES (AA$(,), C$, KY%, MUL$)



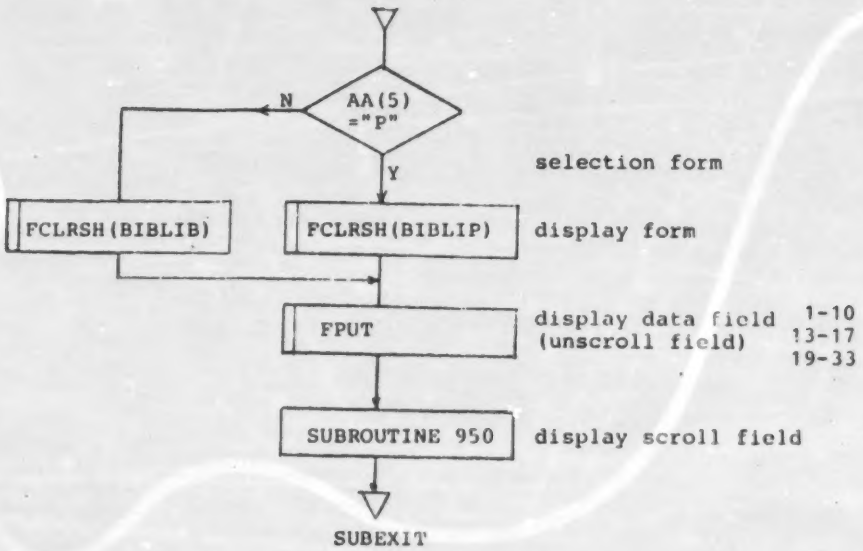
SUBPROGRAM SELECT

PROGRAM NAME: SELECT (MUL$, C$, KY%)



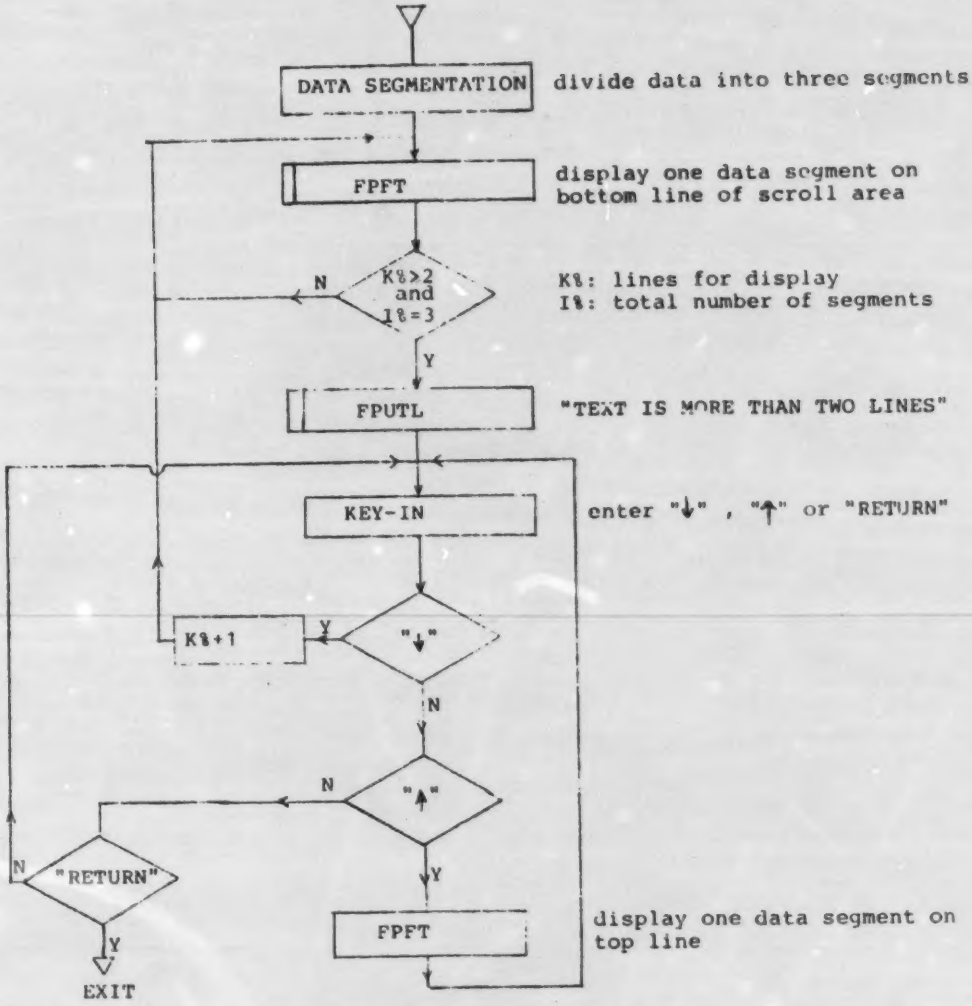
SUBPROGRAM DISPLAY

PROGRAM NAME: DISPLAY (AA$(,))



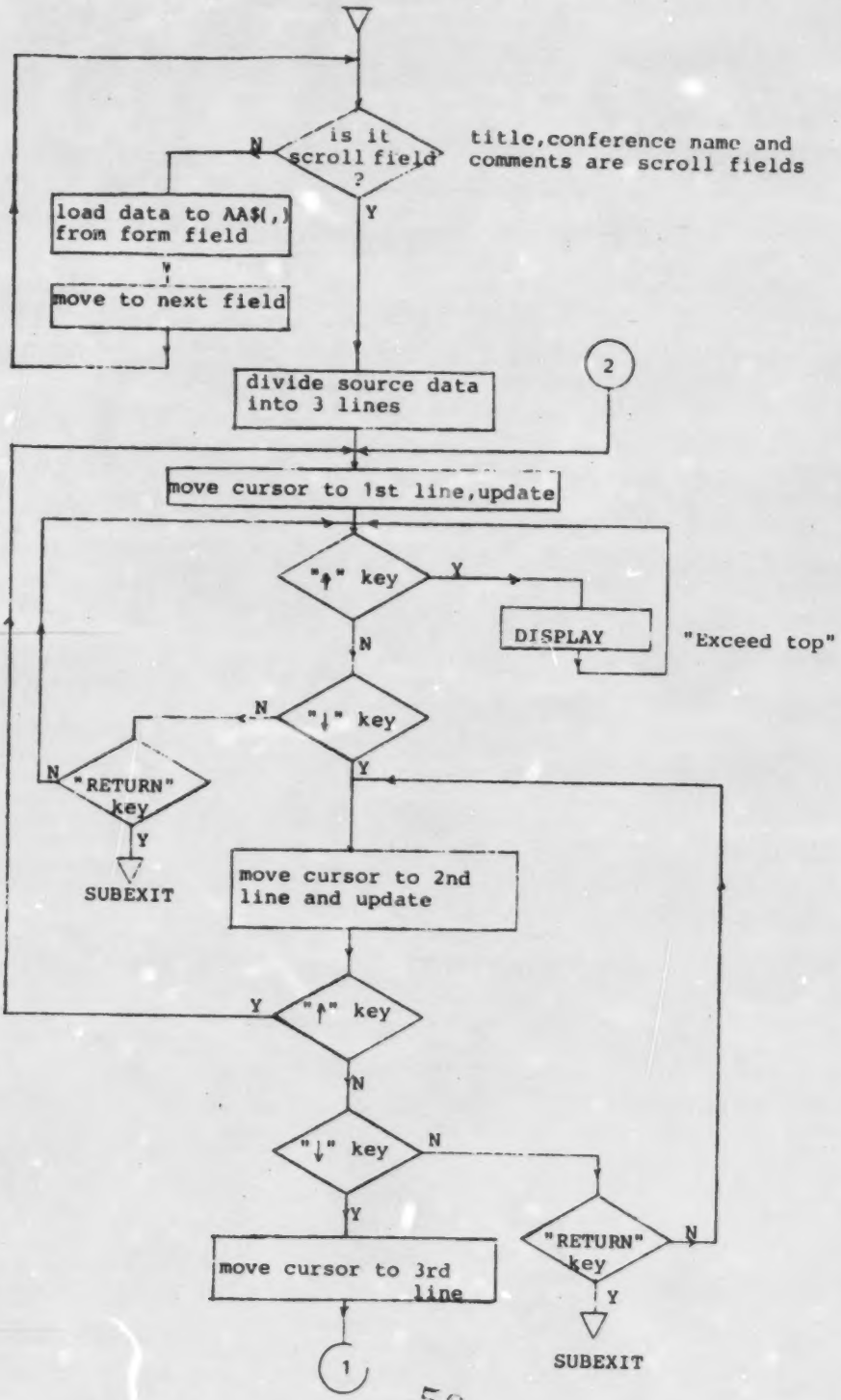
SUBROUTINE (950)

Display of data in scroll area

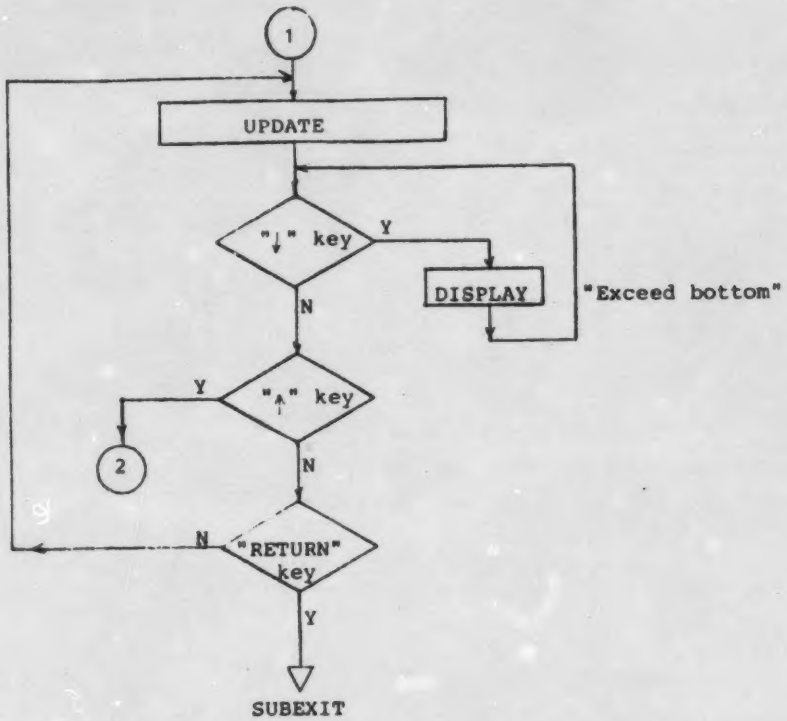


SUBPROGRAM UPDATE

PROGRAM NAME: UPDATE (AA$(,))

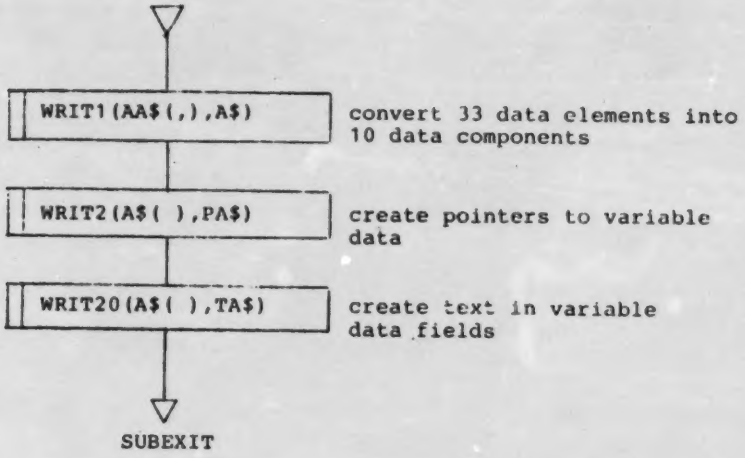


SUBPROGRAM UPDATE (continued)



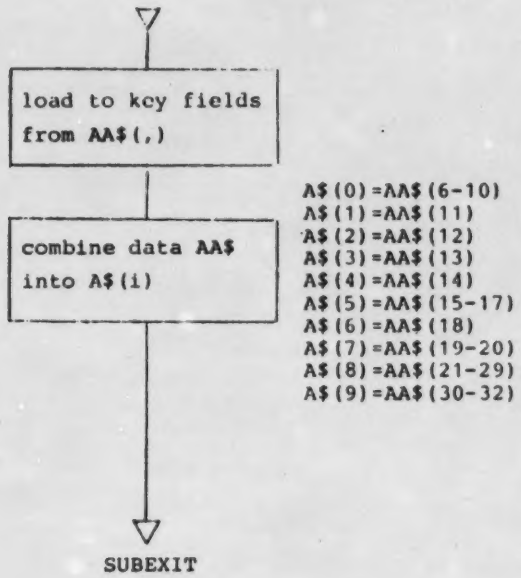
SUBPROGRAM REWRIT

PROGRAM NAME: REWRIT (AA$ (,), PA$, TA$)



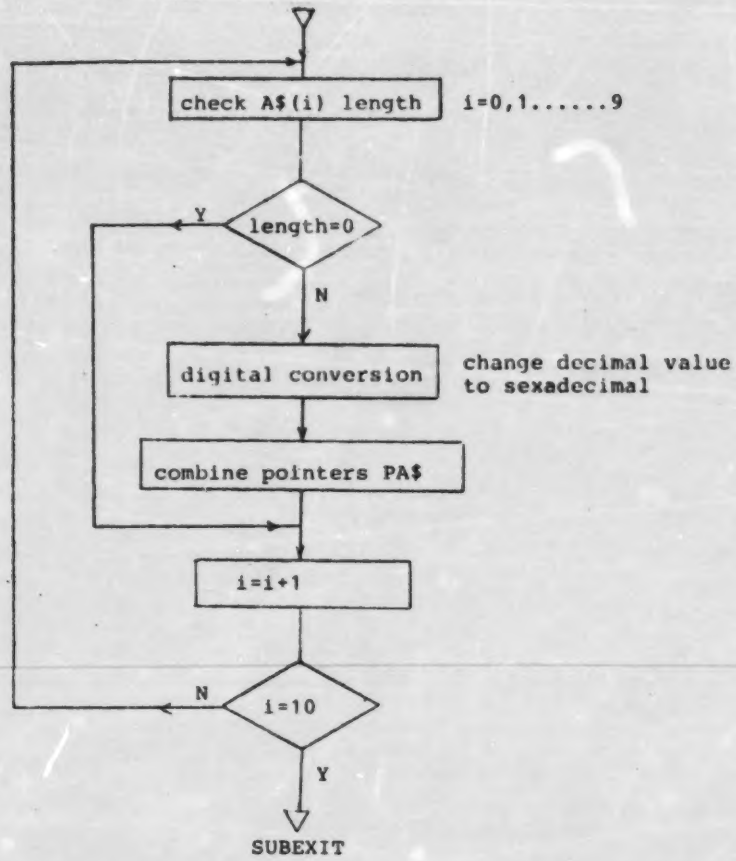
SUBPROGRAM WRIT1

PROGRAM NAME: WRIT1 (AA$ (,), A$ ())



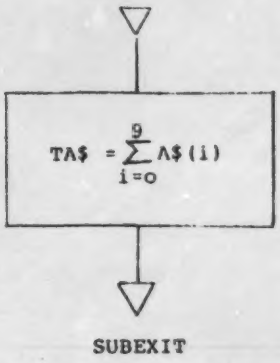
SUBPROGRAM WRIT 2

PROGRAM NAME: WRIT2 (A$ (), PA$)



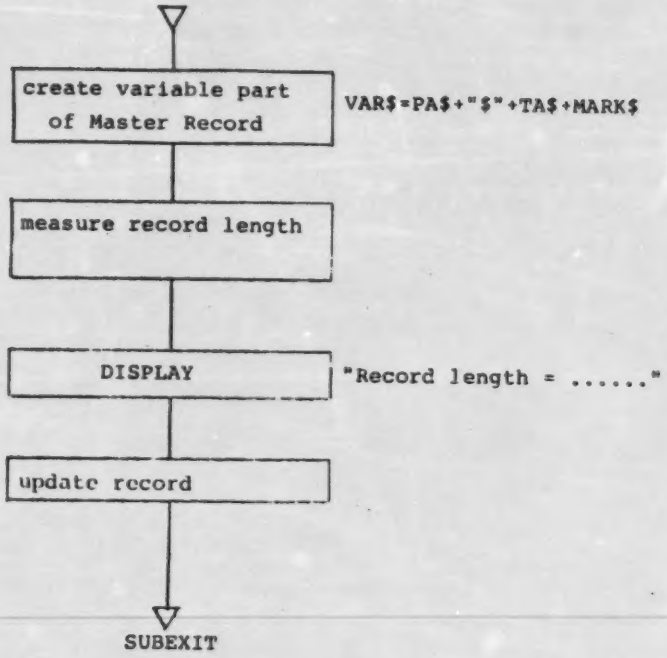
SUBPROGRAM WRIT20

PROGRAM NAME: WRIT20 (A$ (), TA$)

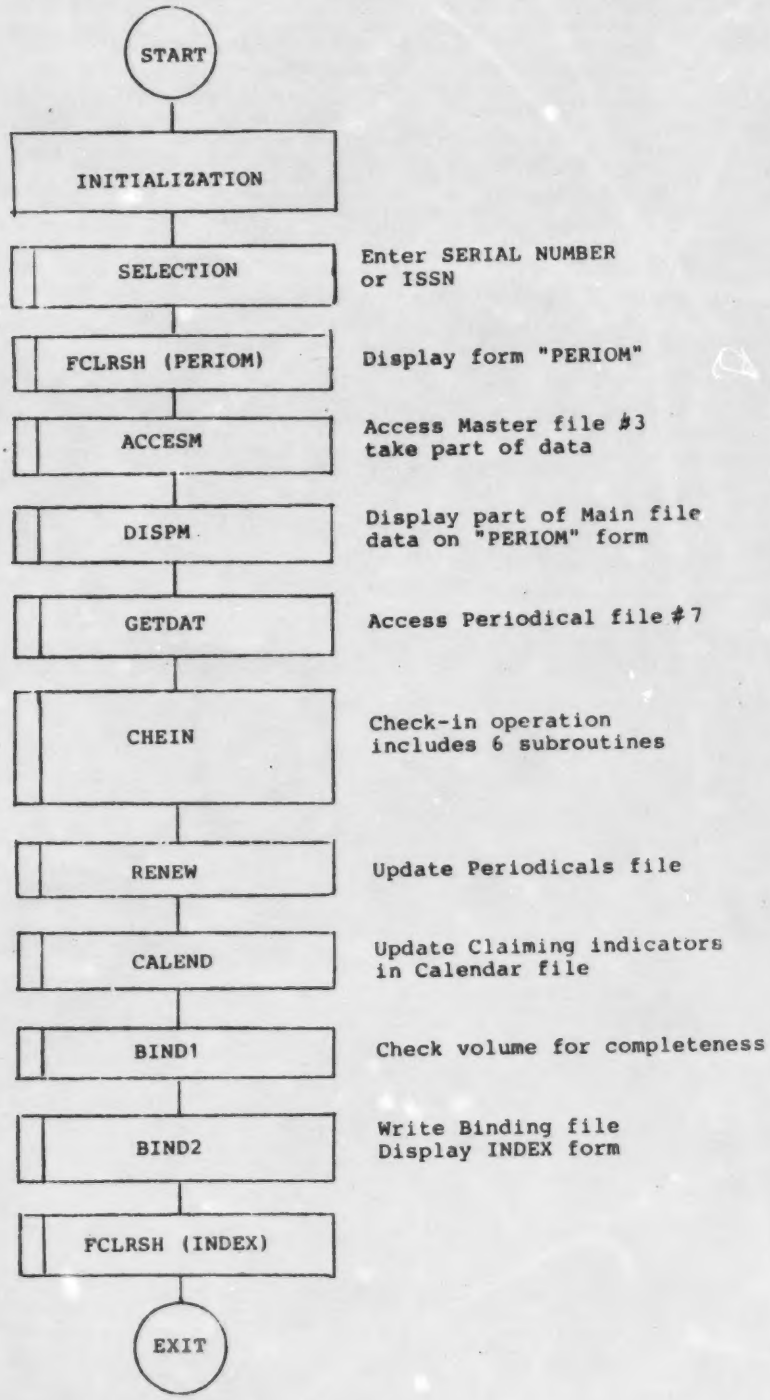


SUBPROGRAM WRIT 3

PROGRAM NAME: WRIT 3 (PA$, TA$, MARK$)

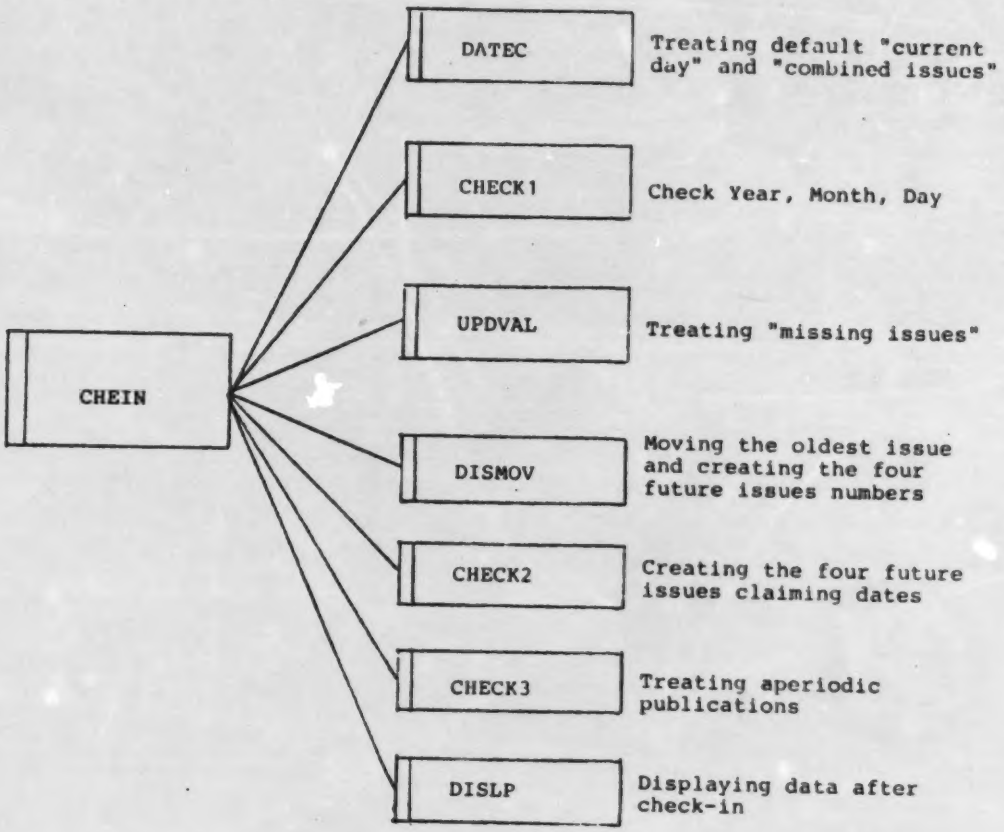


PERIODICALS MANAGEMENT SYSTEM: MAIN PROGRAM



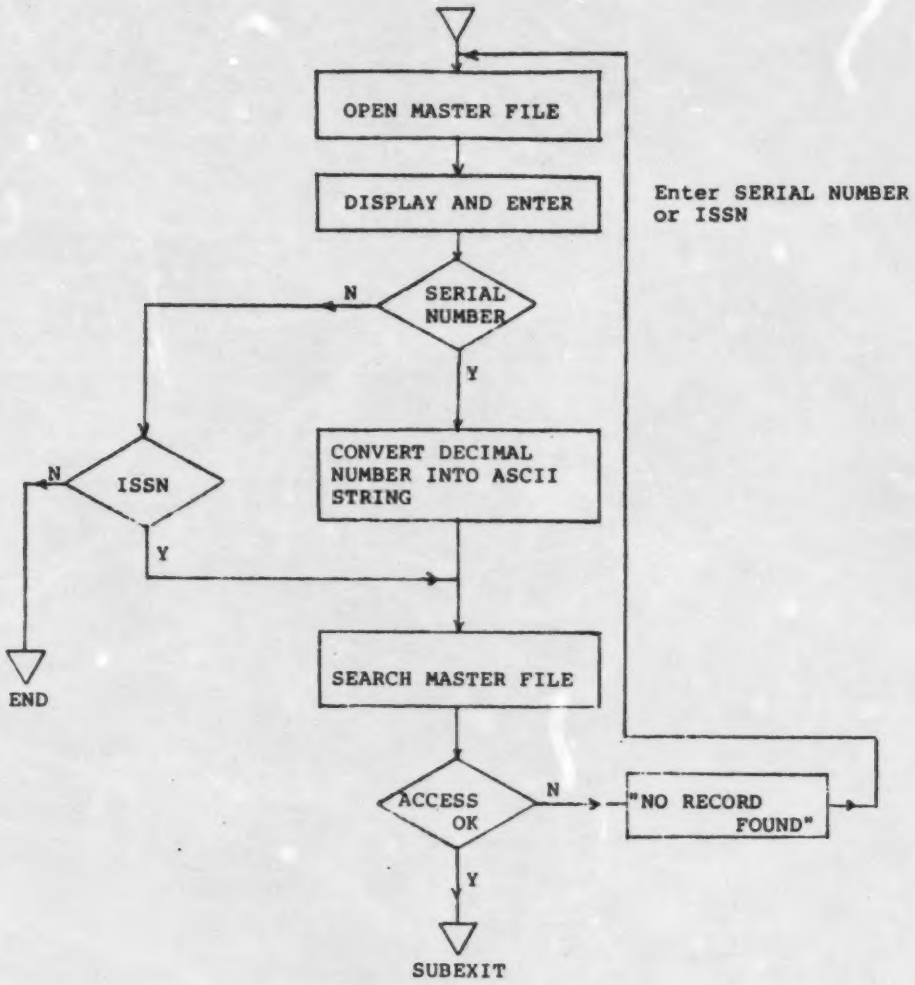
PERIODICALS MANAGEMENT SYSTEM: MAIN PROGRAM

(continued)



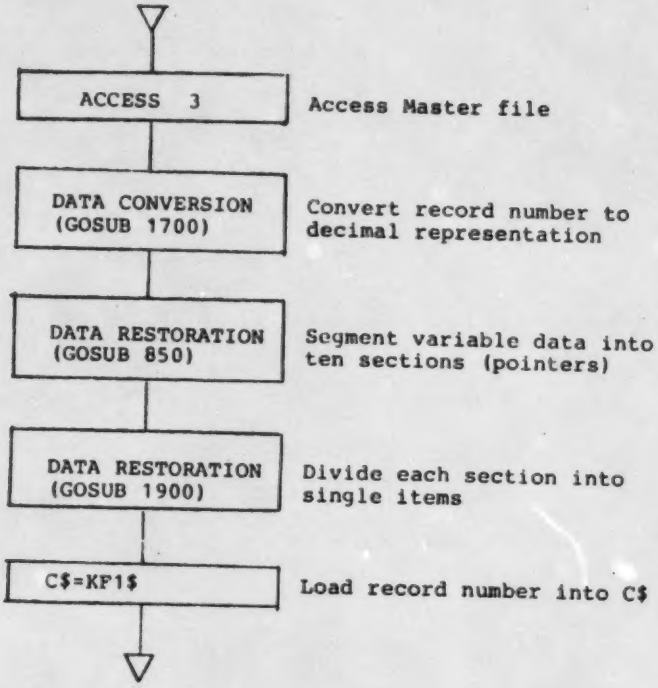
SUBPROGRAM CHOICE

PROGRAM NAME: CHOICE (ERR%)

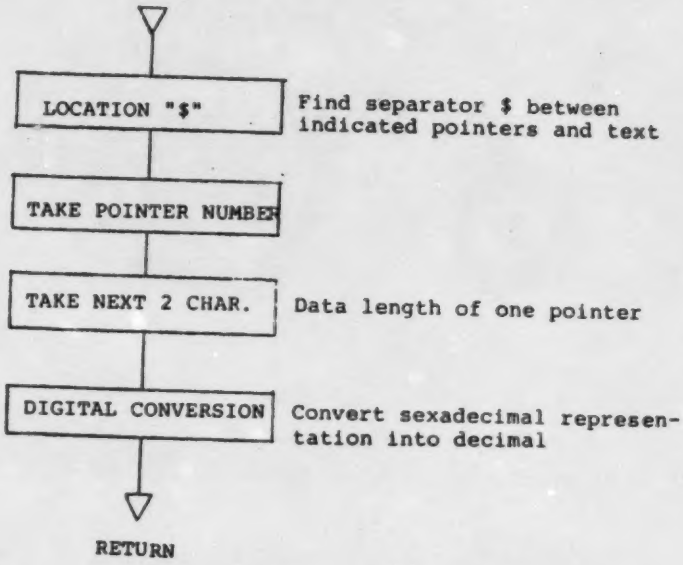


SUBPROGRAM ACCESM

PROGRAM NAME: ACCESM (AA$ ( ), C$)

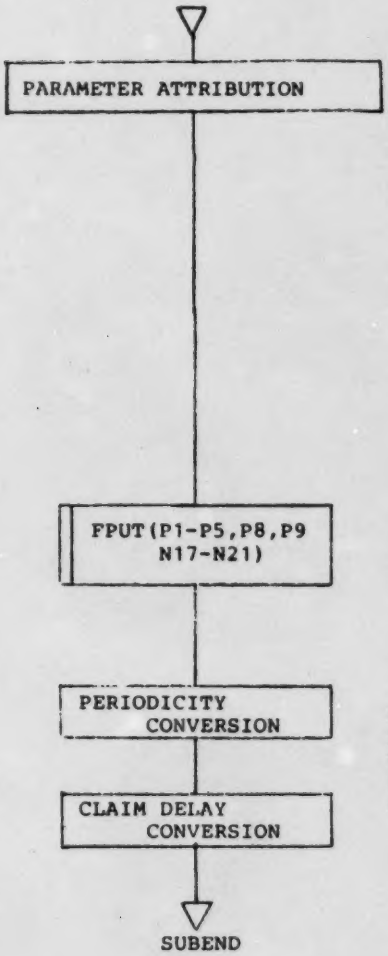

ACCESS 3
Access Master file
DATA CONVERSION (GOSUB 1700)
Convert record number to decimal representation
DATA RESTORATION (GOSUB 850)
Segment variable data into ten sections (pointers)
DATA RESTORATION (GOSUB 1900)
Divide each section into single items
C$=KF1$
Load record number into C$

SUBROUTINE OF VARIABLE RESTORATION (GOSUB 850)



SUBPROGRAM DISPM

PROGRAM NAME: DISPM (AA$ (,), N$ ( ))



Periodicity:

d=1 q=91
c=4 t=122
w=7 f=183
e=14 a=365
s=15 g=730
m=30 h=1095
b=60

Claim delay:

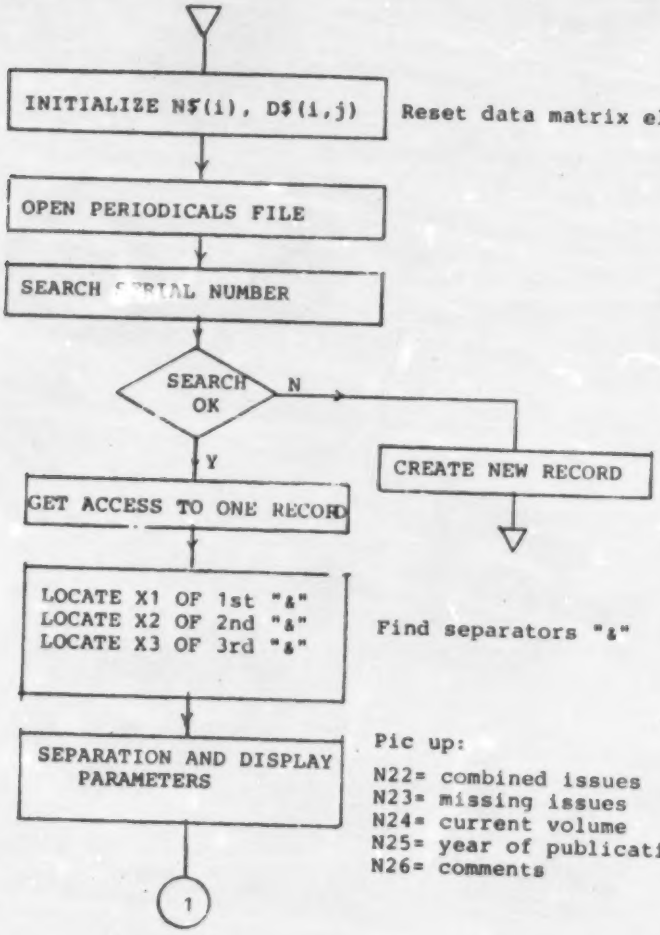
h=15 d=60
t=21 q=90
a=30 s=180
f=45

Display:

P1=RECORD NUMBER
P2=ISSN
P3=SIGNUM
P4=LOCATION RITL
P5=SEPARATE CATALOGUING
P8=COUNTRY
P9=IMPRESSUM
N17=PERIODICITY
N18=NUMBER/VOL
N19=INDEX MODE
N20=CLAIM MODE
N21=CLAIM DELAY

SUBPROGRAM GETDAT

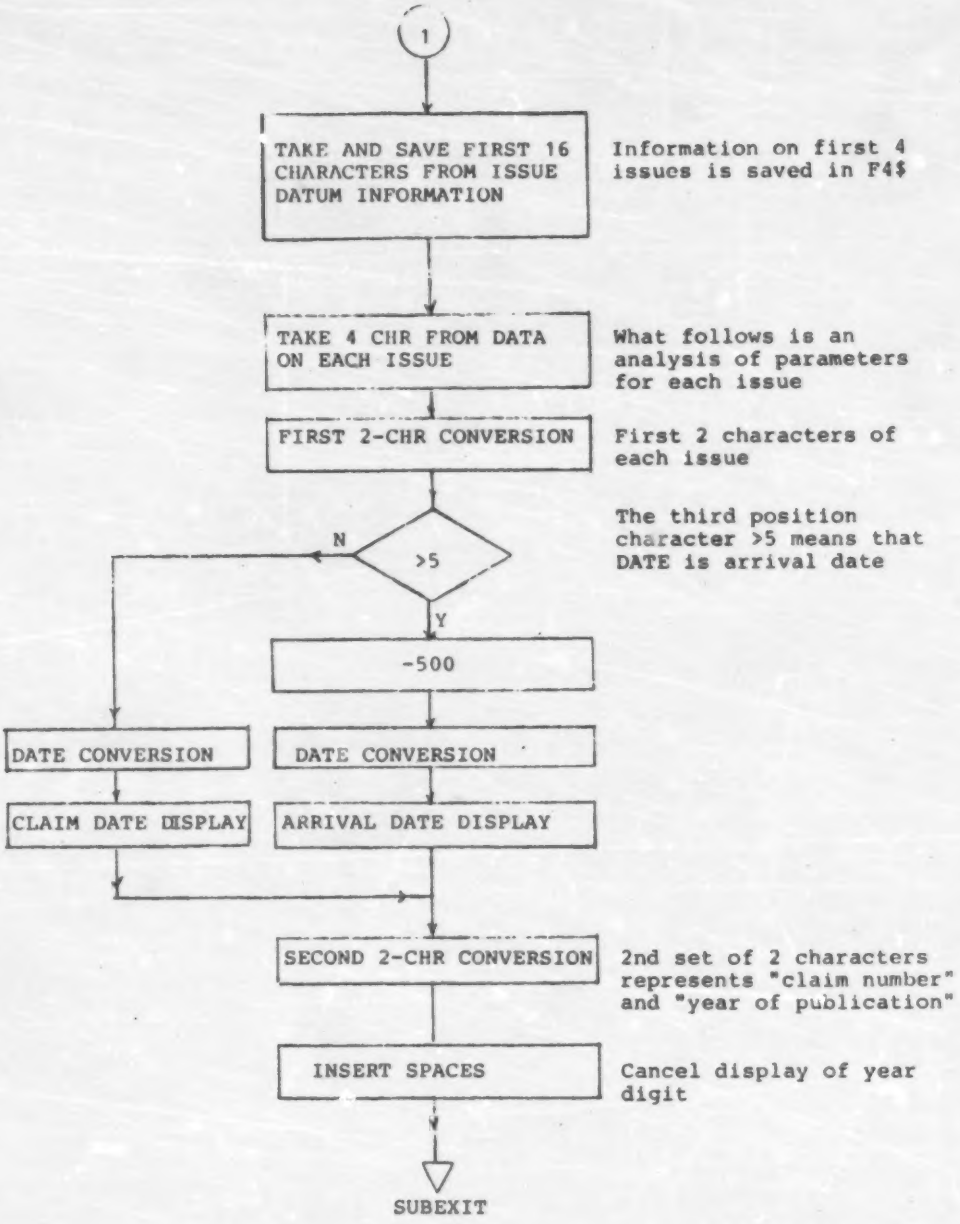
PROGRAM NAME: GETDAT (D$ (,), N$ ( ), C$, CLAIM, F4$)



DATA FORMAT IN PERIODICALS FILE:

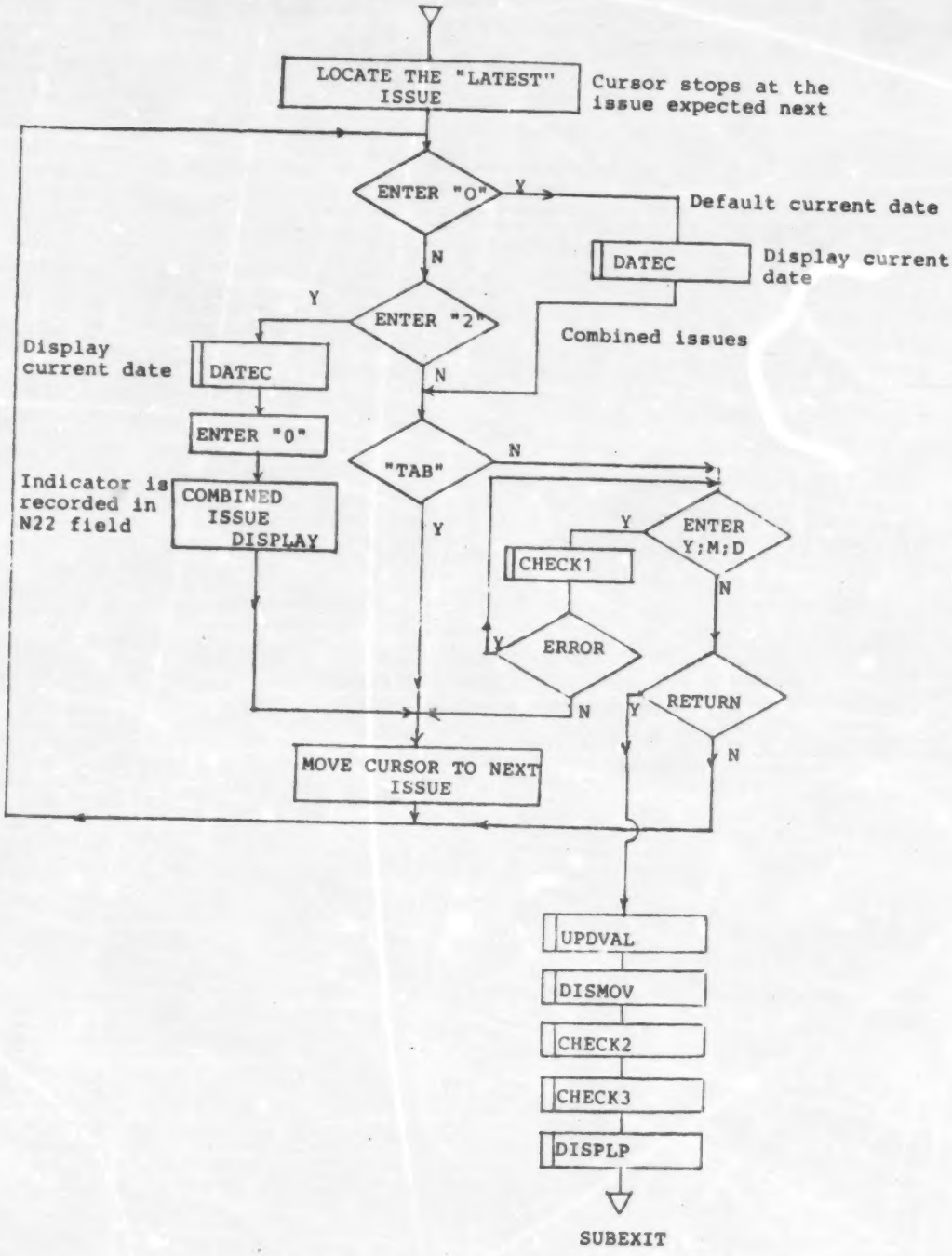
xxx	aaa	bb	&	ccc...cc	&	ddd...dd	&	eee...ee	&	eee...eee...
rec.No	N24	N25	X1	N22		N23	X2	N26	X3	issue data information

SUBPROGRAM GETDAT (continued)


1
TAKE AND SAVE FIRST 16 CHARACTERS FROM ISSUE DATUM INFORMATION
Information on first 4 issues is saved in F4$
TAKE 4 CHR FROM DATA ON EACH ISSUE
What follows is an analysis of parameters for each issue
FIRST 2-CHR CONVERSION
First 2 characters of each issue
N
>5
The third position character >5 means that DATE is arrival date
Y
-500
DATE CONVERSION
DATE CONVERSION
CLAIM DATE DISPLAY
ARRIVAL DATE DISPLAY
SECOND 2-CHR CONVERSION
2nd set of 2 characters represents "claim number" and "year of publication"
INSERT SPACES
Cancel display of year digit
SUBEXIT

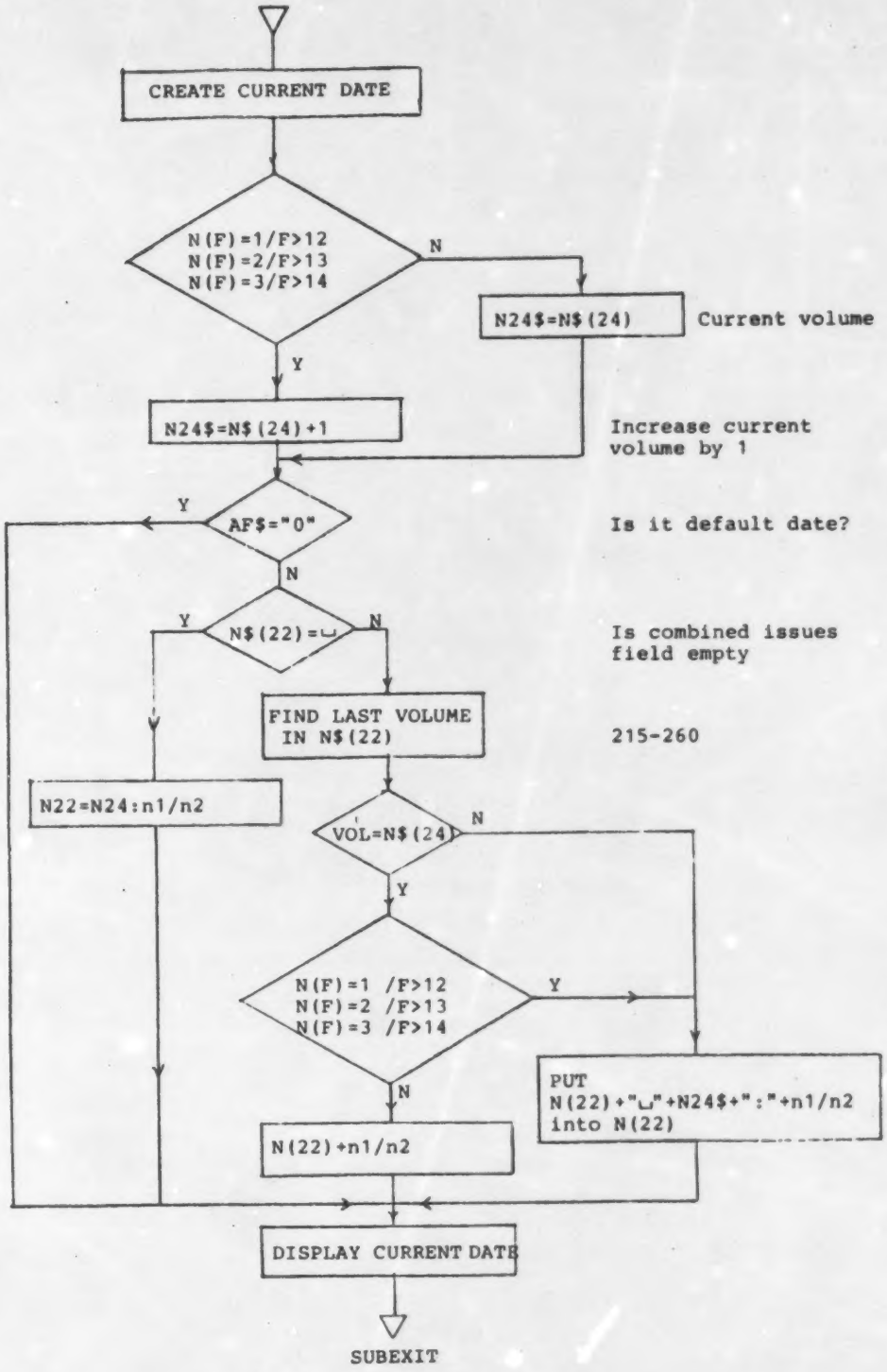
SUBPROGRAM CHEIN

PROGRAM NAME: CHEIN (D$ (,), N$ ( ), J %, VOL$, Z%)



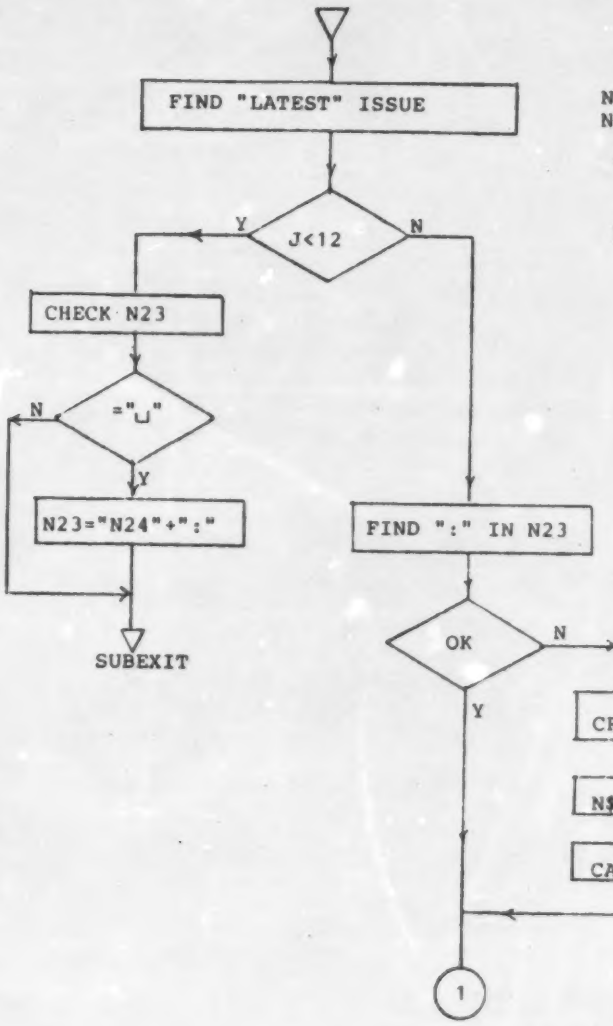
SUBPROGRAM DATEC

PROGRAM NAME: DATEC (AF$, FID$, M2$ ( ), M3$ ( ), N$ ( ))



SUBPROGRAM UPDVAL

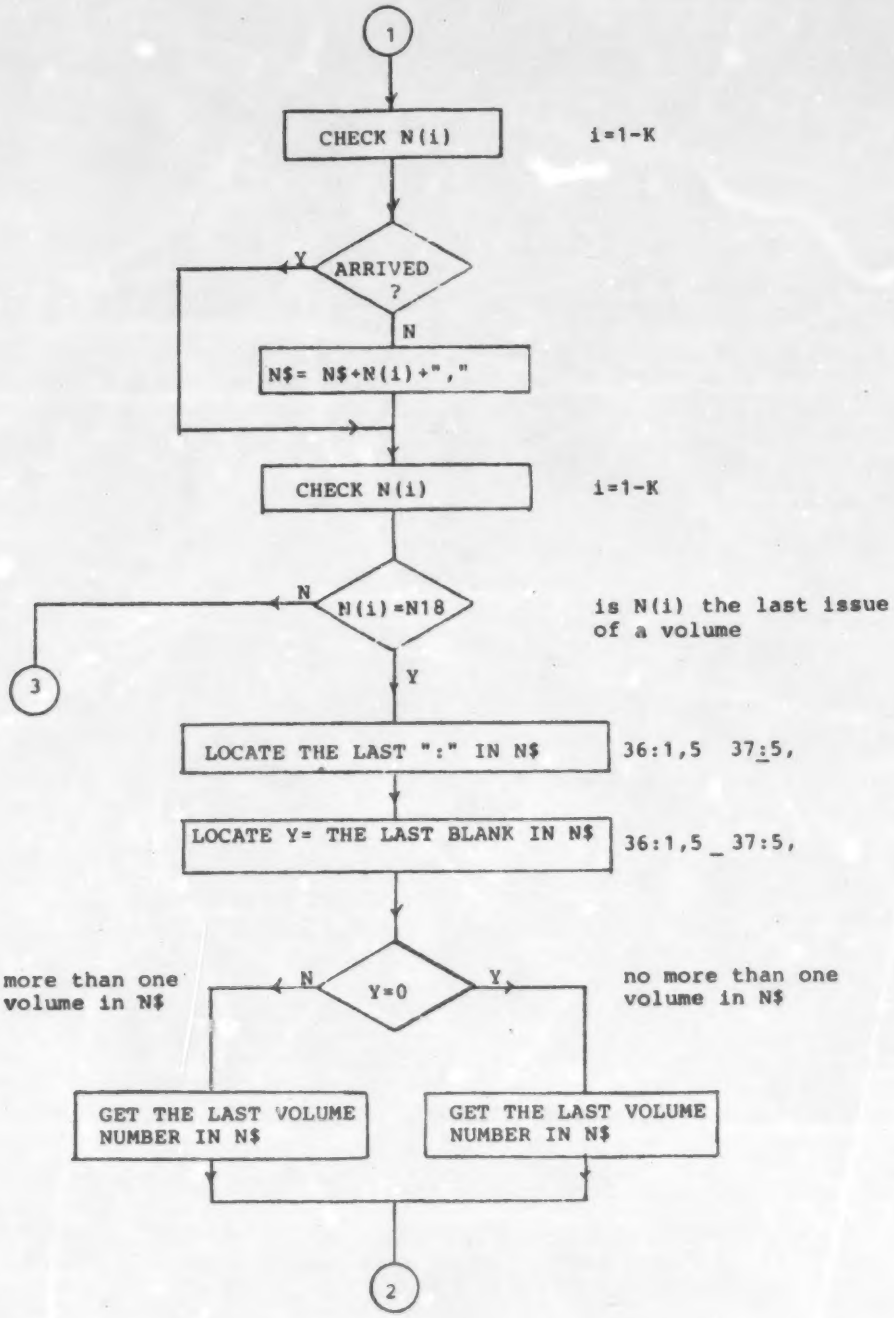
PROGRAM NAME: UPDVAL (D$ (,), N$ ( ), J%)



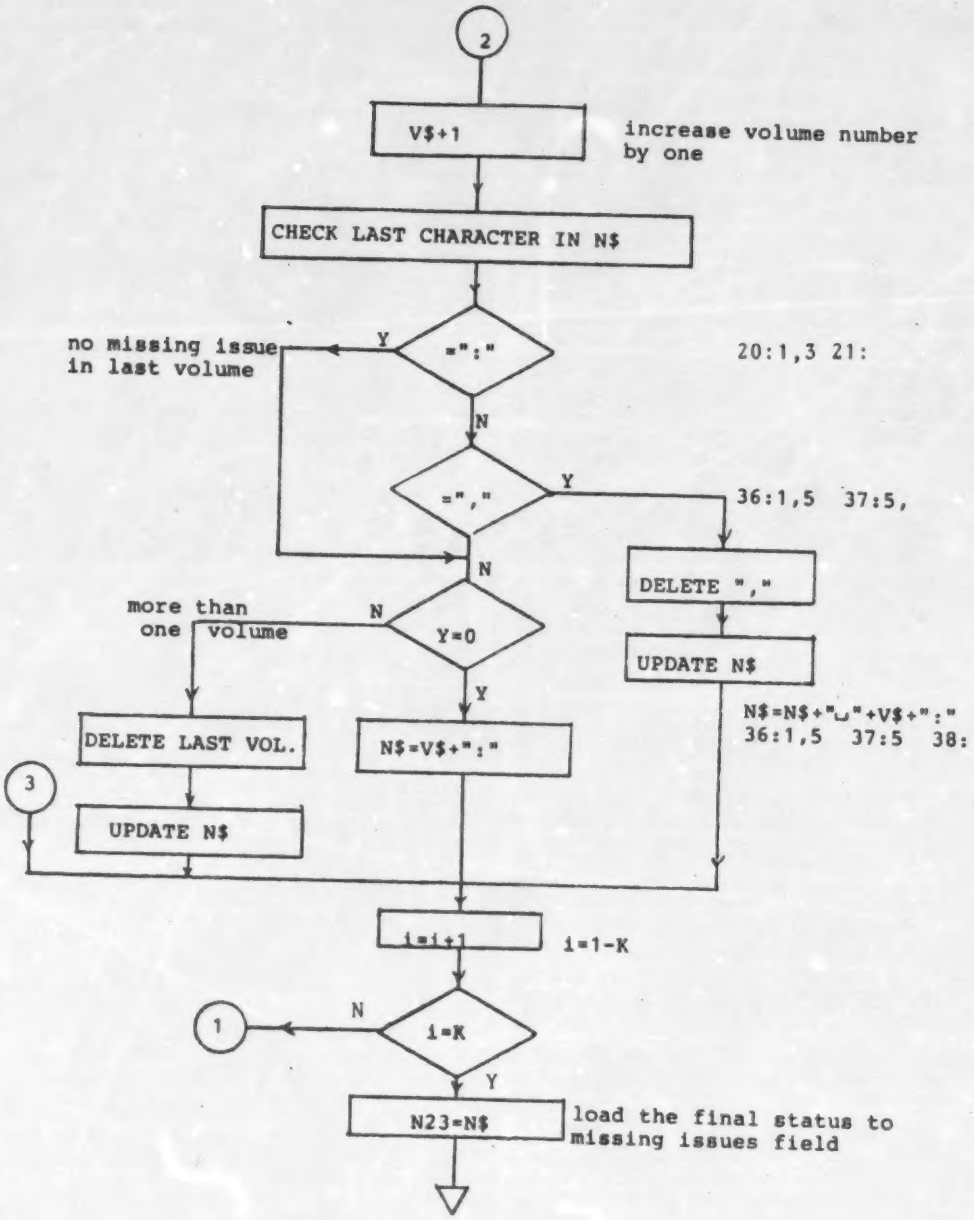
N23: missing issues field
N24: current volume field
J: "latest" issue number
V$: corresponds to the volume number of the first issue displayed
N$: missing information
K: number to be moved out of the screen

find 1st issue of volume concerned

SUBPROGRAM UPDVAL (continued)

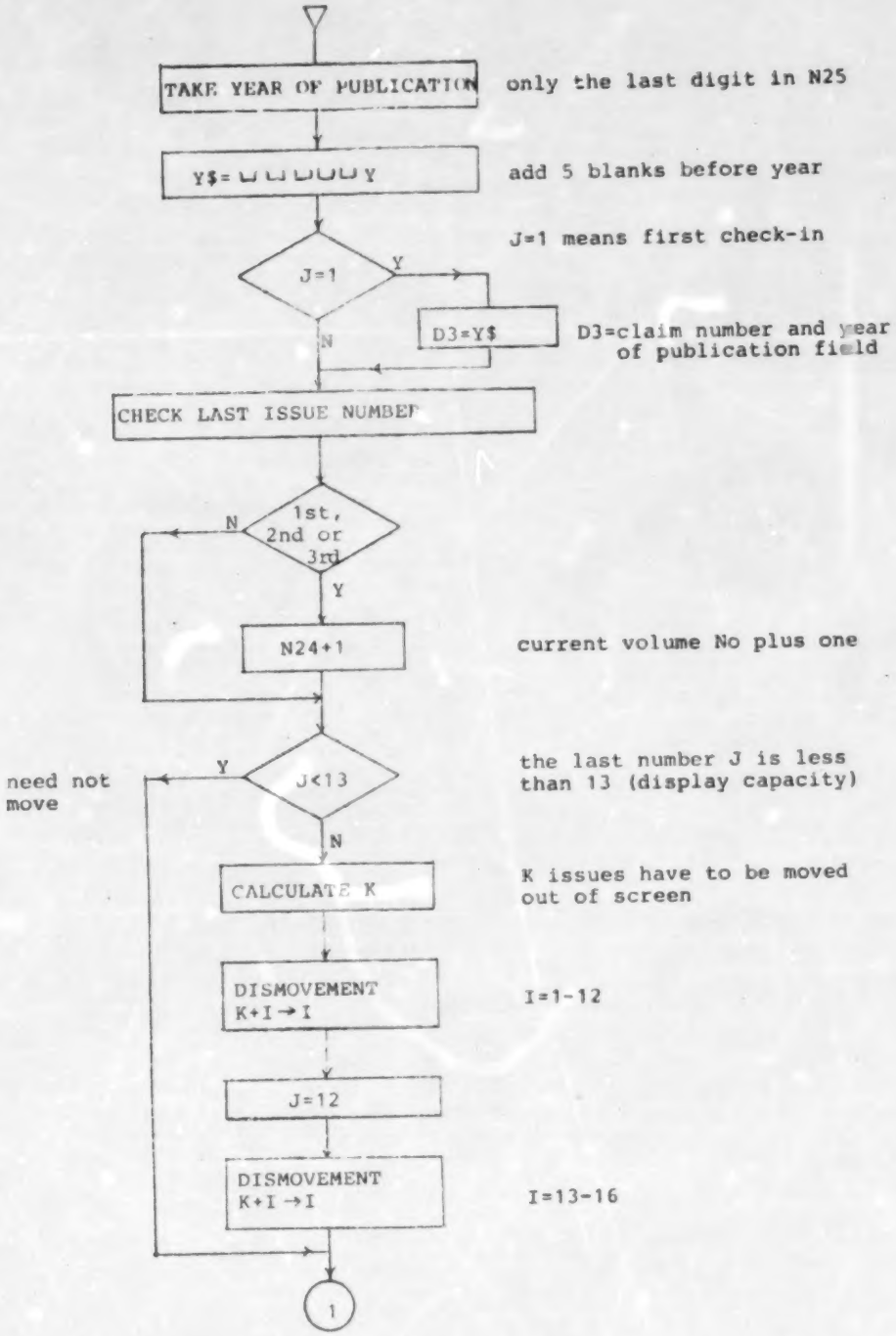


SUBPROGRAM UPDVAL (continued)

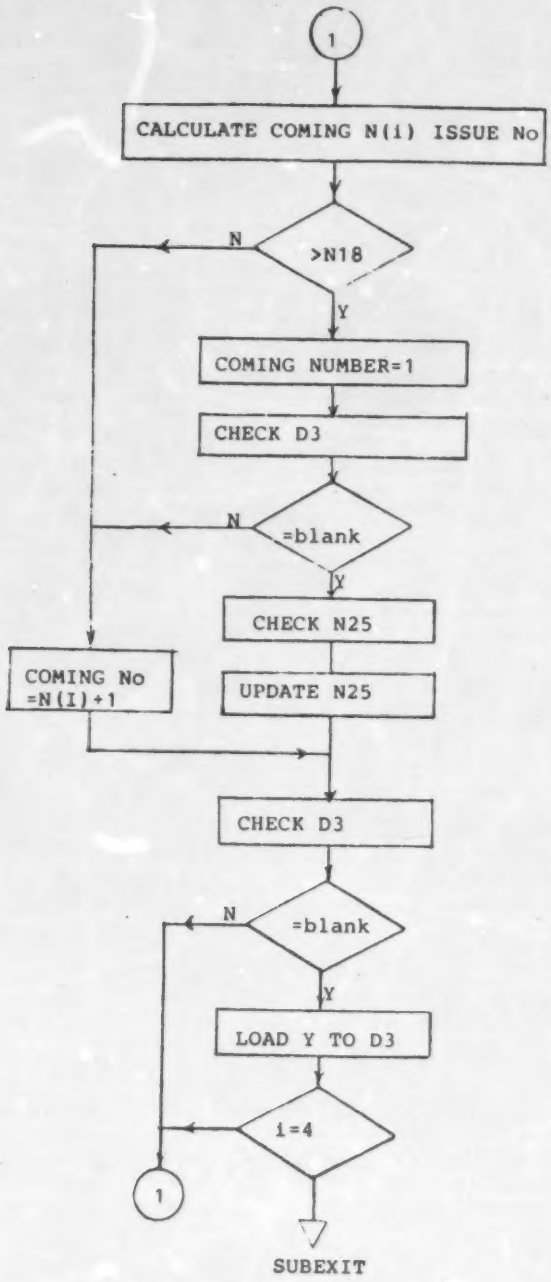


SUBPROGRAM DISMOV

PROGRAM NAME: DISMOV (D$(,), N$( ), J%, Z%)

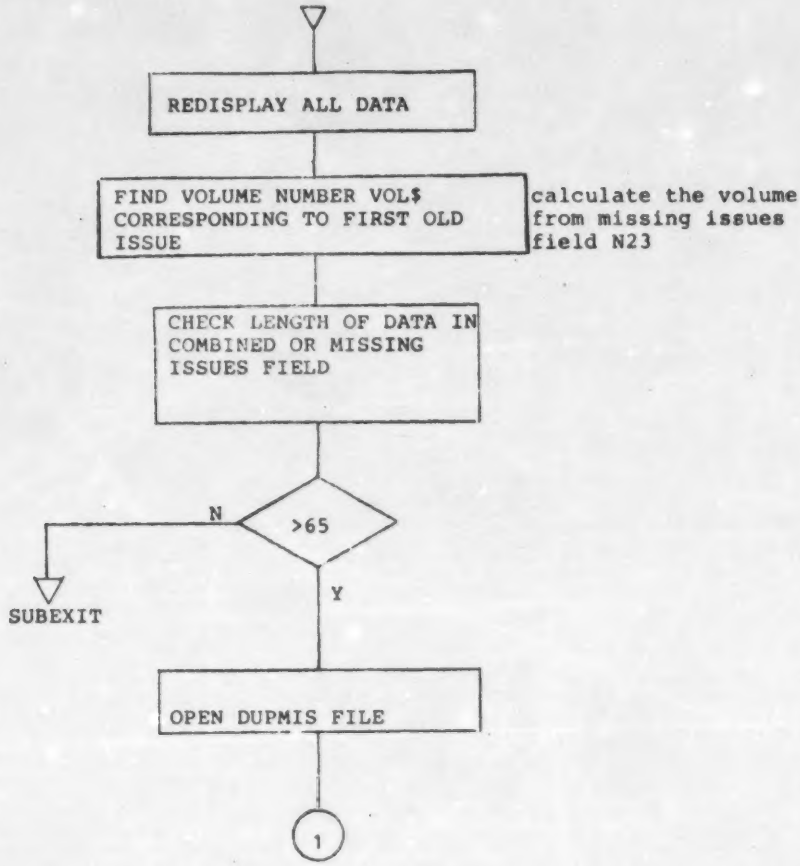


SUBPROGRAM DISMOV (continued)



SUBPROGRAM DISPLP

PROGRAM NAME: DISPLP (D$(,), J%, VOL$)

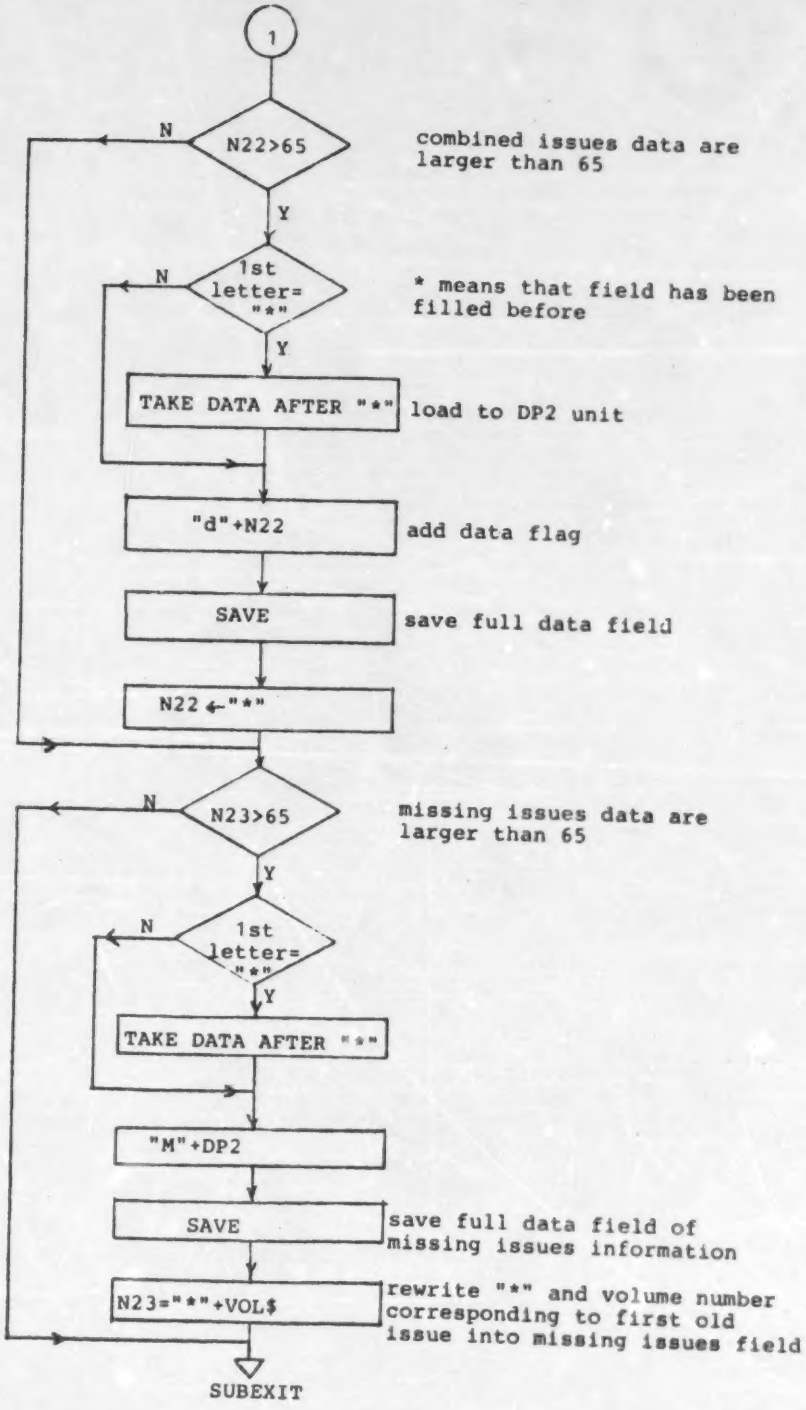


DATA FORMAT IN DUPMIS FILE:

XXX F YYYYY.....Y
1 2 3

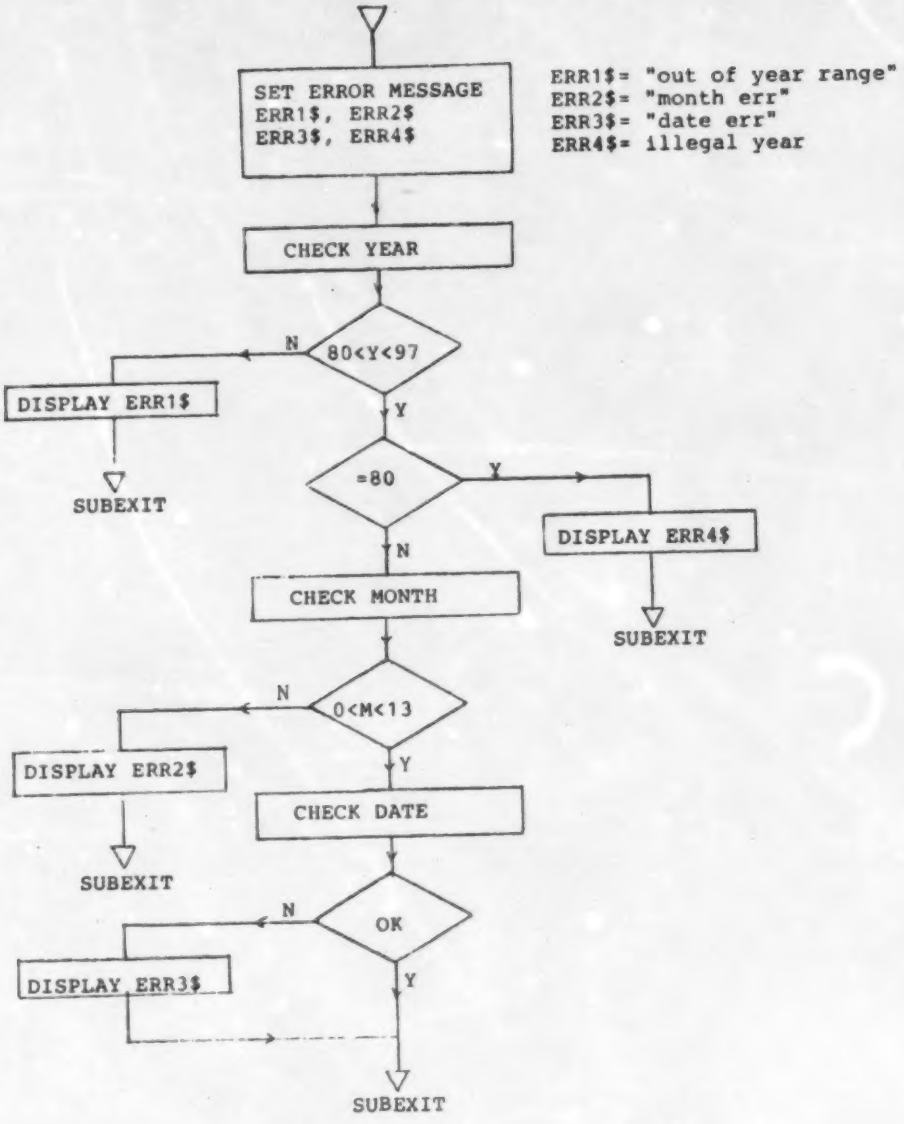
1= 3-digit record number
2= 1-digit data flag: D represents data on combined issue
M represents data on missing issue
3= data field

SUBPROGRAM DISPLP (continued)



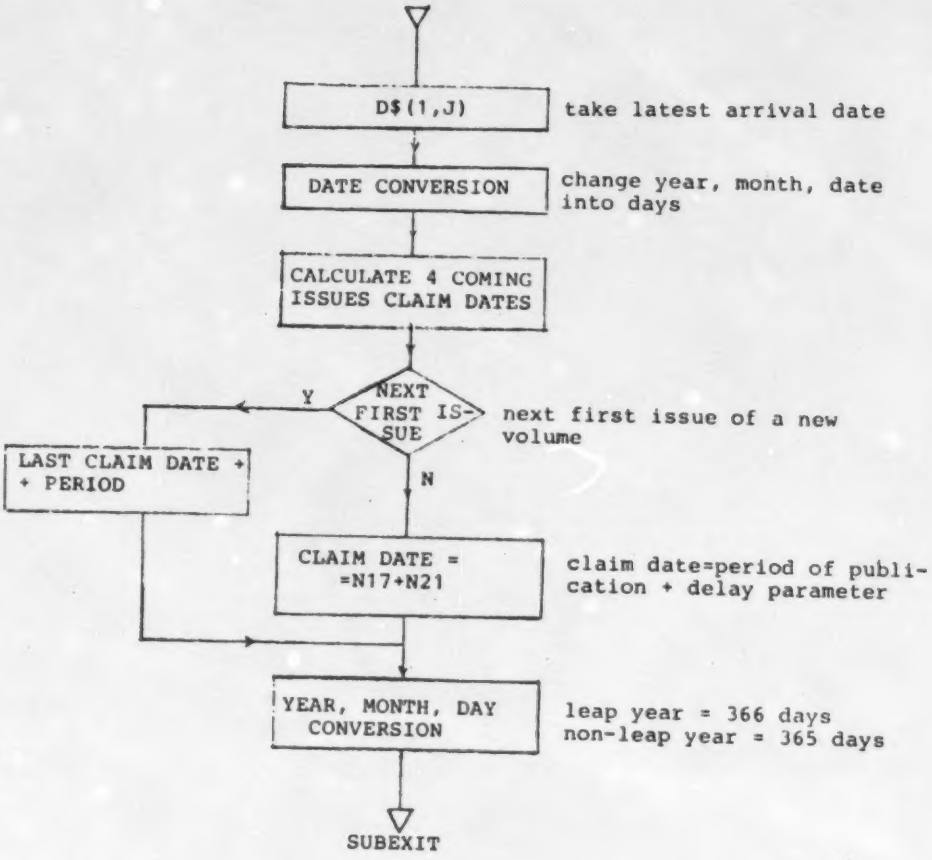
SUBPROGRAM CHECK 1

PROGRAM NAME: CHECK 1 (YMD$, ER%, M1 ( ), M3$ ( ))



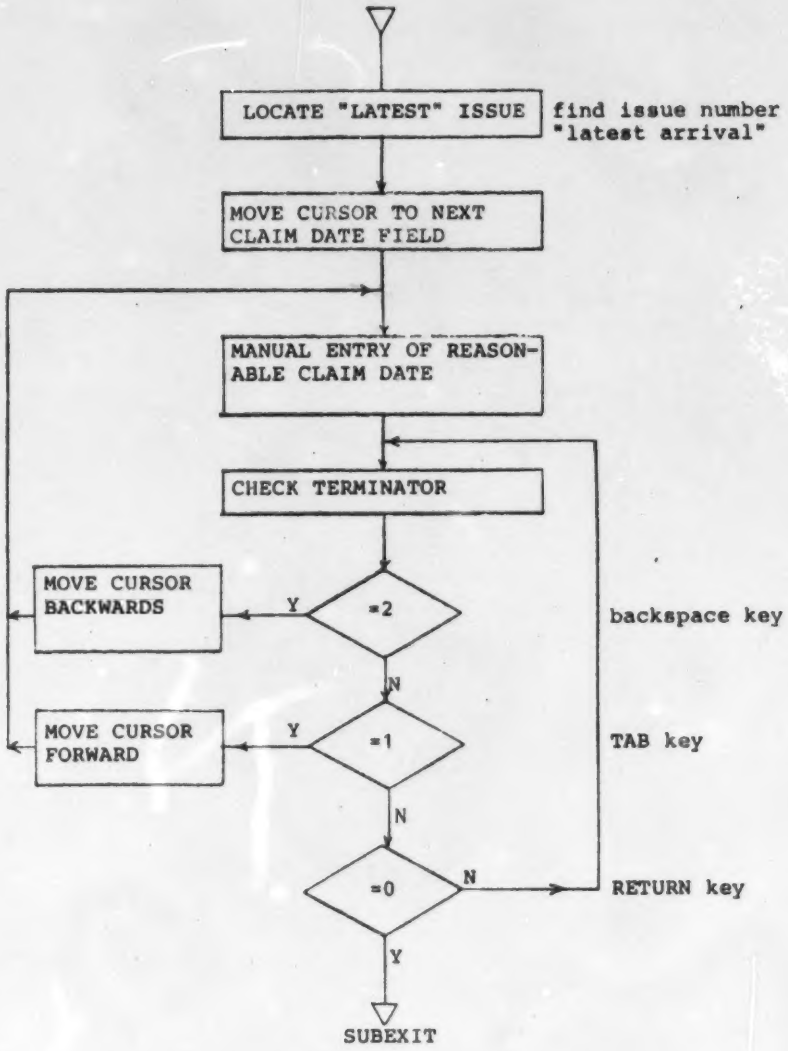
SUBPROGRAM CHECK 2

PROGRAM NAME: CHECK 2



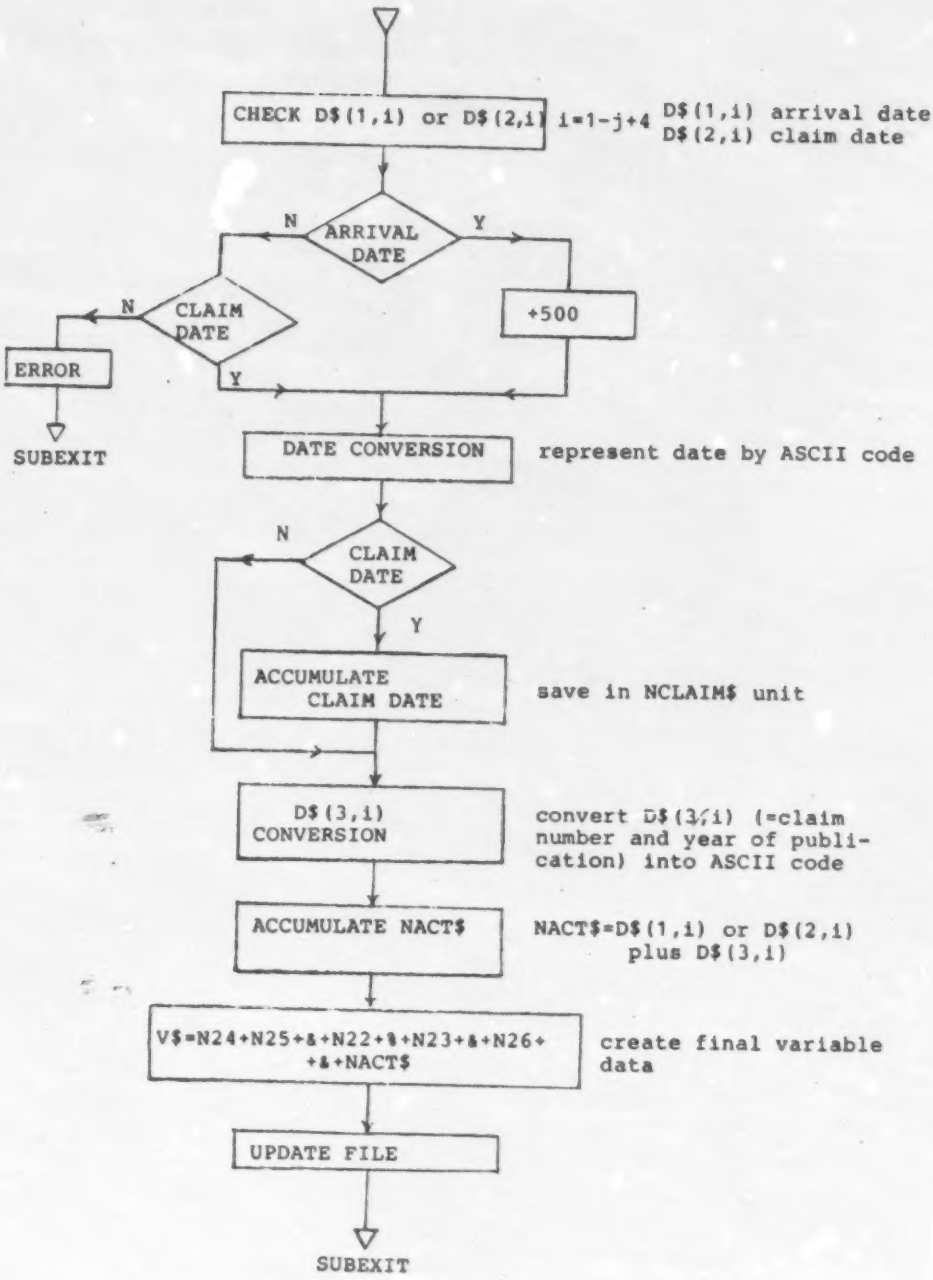
SUBPROGRAM CHECK 3

PROGRAM NAME:CHECK 3 (D$ (,), N$ ( ))



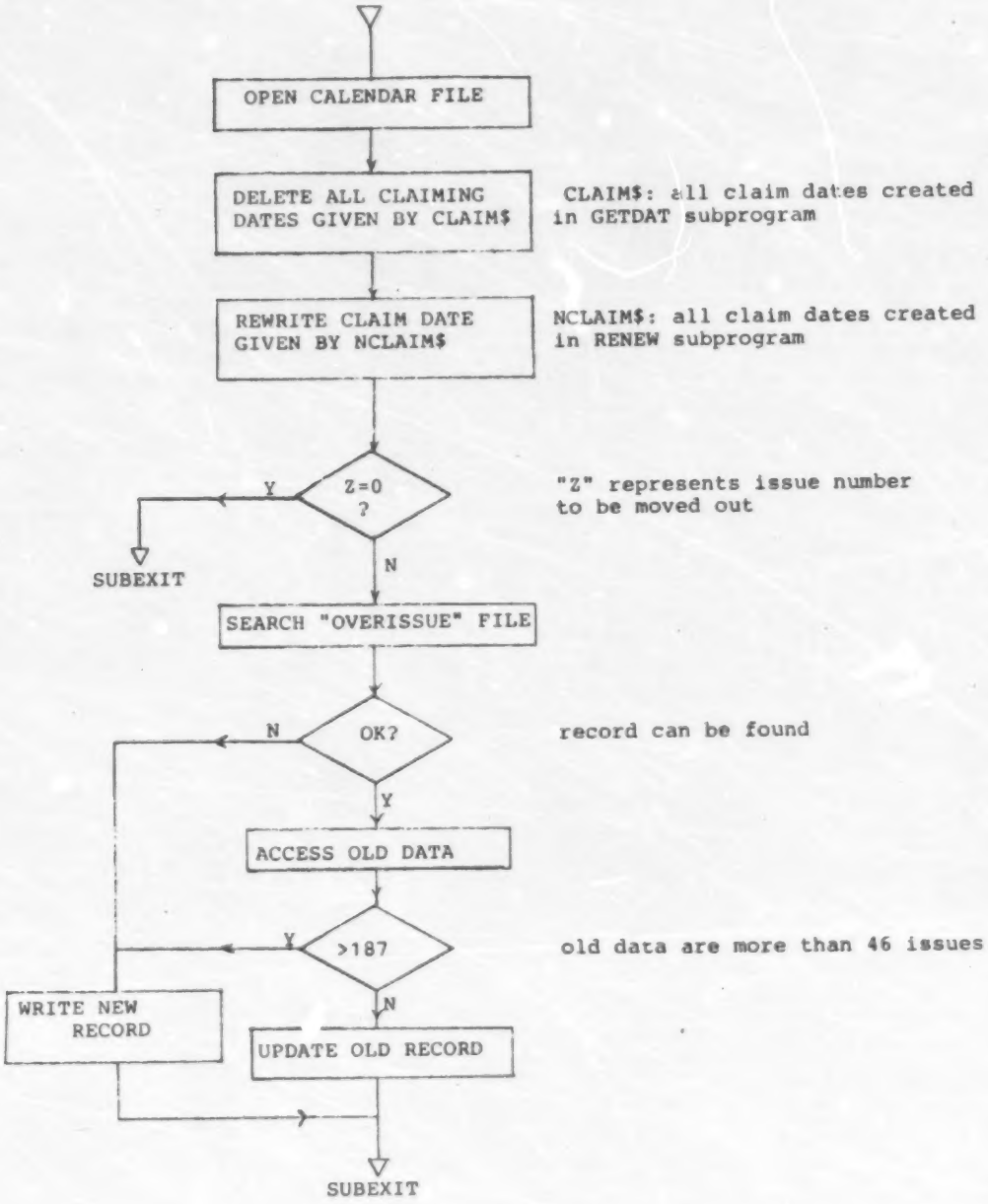
SUBPROGRAM RENEW

PROGRAM NAME: RENEW (D$ (,), N$ ( ), J%, C$, NCLAIM$, VOL$)



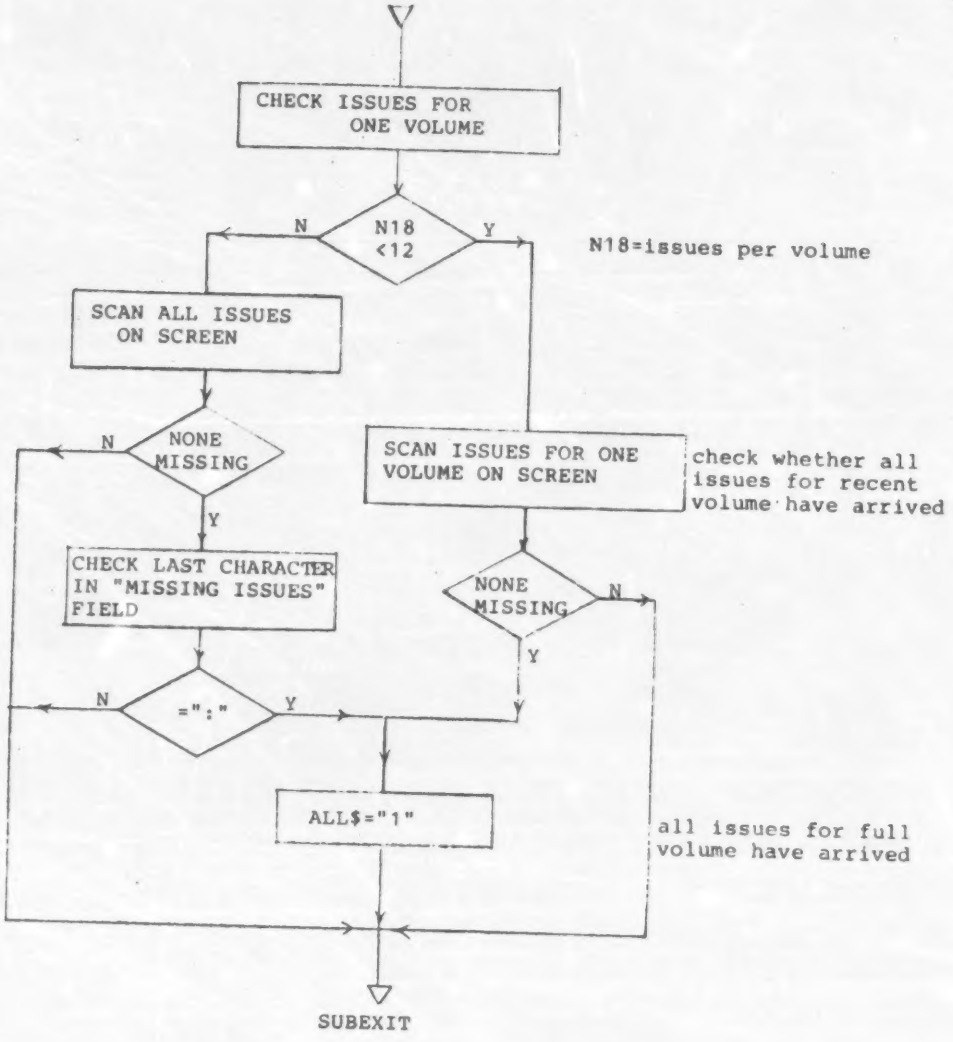
SUBPROGRAM CALEND

PROGRAM NAME: CALEND (CLAIM$, NCLAIM$, F4$, Z%)



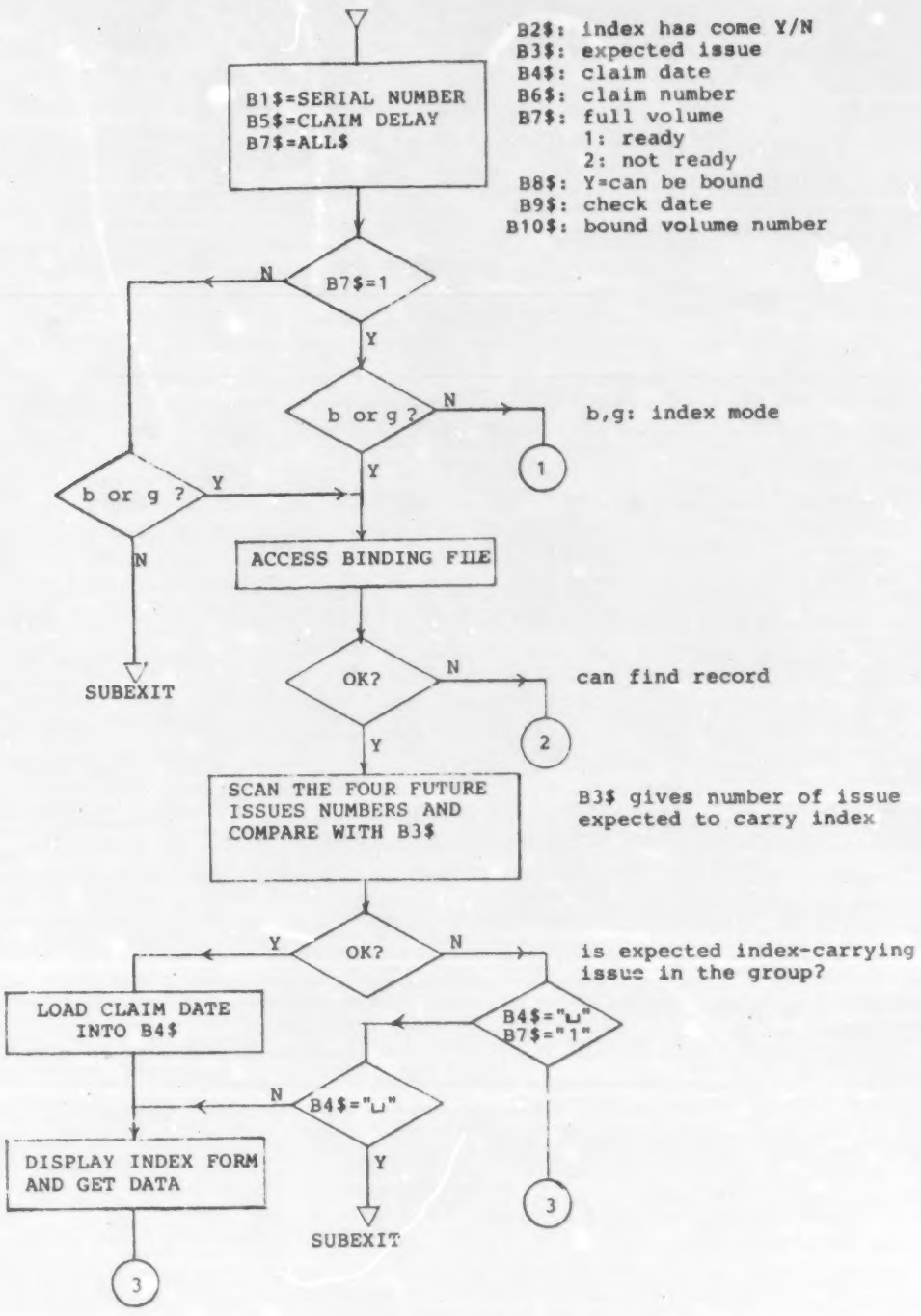
SUBPROGRAM BIND 1

PROGRAM NAME:BIND 1 (D$ (,), N$ ( ), J%, ALL$)

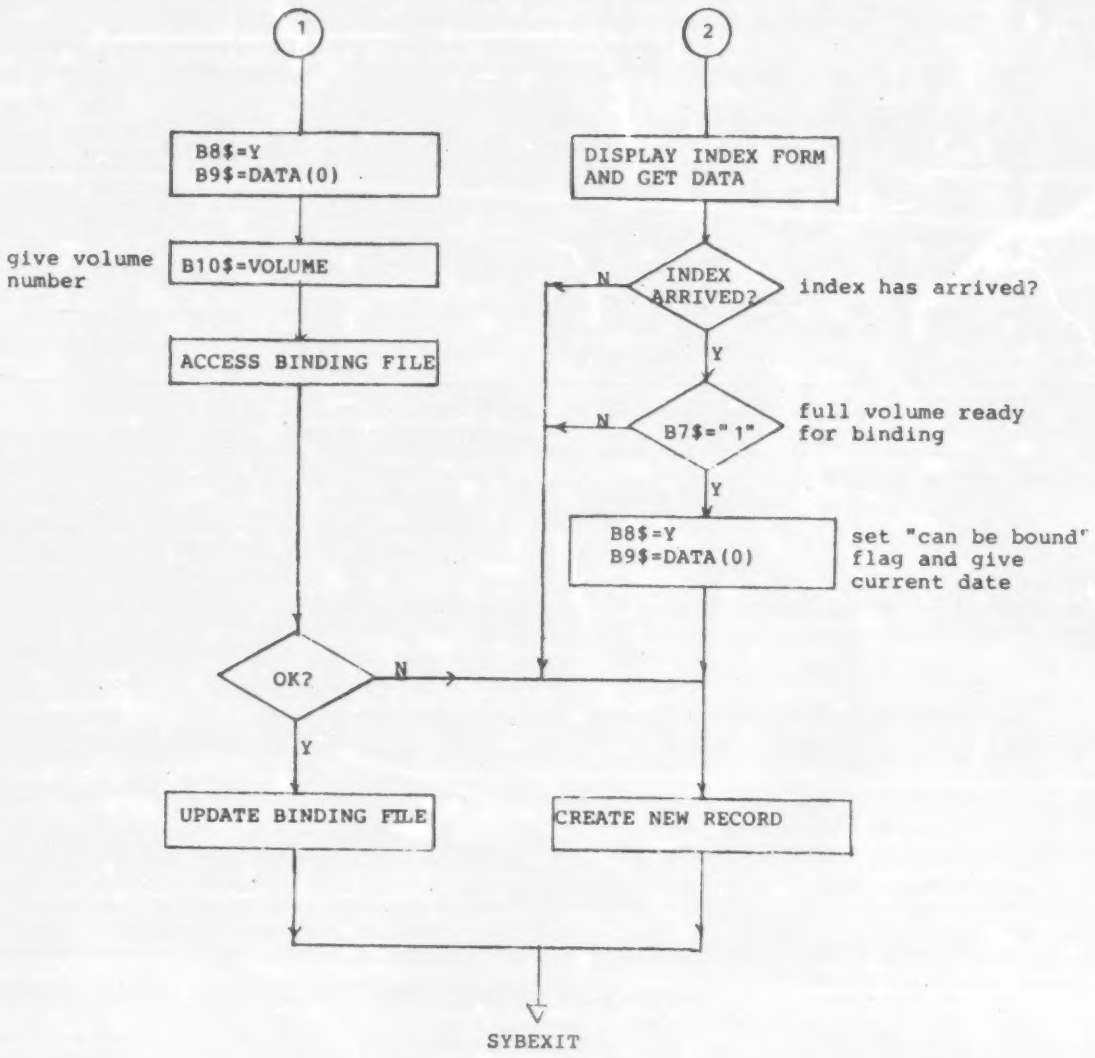


SUBPROGRAM BIND 2

PROGRAM NAME: BIND 2 (D$ (,), N$ ( ), J%, ALL$)



SUBPROGRAM BIND 2 (continued)



SUBPROGRAM BIND 2 (continued)